

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ім. Ю. М. Потебні

**Кафедра електроніки, інформаційних систем
та програмного забезпечення**
(повна назва кафедри)

Кваліфікаційна робота
другий (магістерський)
(рівень вищої освіти)

на тему : Дослідження та розробка мікроелектронної інформаційної системи
проведення спортивних змагань

Виконав: студент II курсу, групи 8.1764ДН
спеціальності 176 Мікро- та наносистемна
техніка
(код і назва спеціальності)
освітньої програми 176 Мікроелектронні
інформаційні системи
(код і назва освітньої програми)
Шарапов Ілля Сергійович
(ініціали та прізвище)

Керівник професор кафедри ЕІСПЗ, професор,
к.т.н., Ніконова Зоя Андріївна
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Рецензент інженер-конструктор «Конструкторське
бюро Колосова» Григор'єва І.К.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Запоріжжя

2026

**МІНІСТЕРСТВО СВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ім. Ю.М. Потебні**

Кафедра електроніки, інформаційних систем та програмного забезпечення
Рівень вищої освіти другий (магістерський)
(перший (бакалаврський) рівень)
Спеціальність 176 Мікро- та наносистемна техніка
(назва)
Освітня програма Мікроелектронні інформаційні системи
(шифр)

ЗАТВЕРДЖУЮ

Завідувач кафедри Т. В. Критська

« » 2025 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТОВІ (СТУДЕНТЦІ)

Шарапову Іллі Сергійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи (проєкту) Дослідження та розробка мікроелектронної інформаційної системи проведення спортивних змагань

керівник роботи Ніконова З.А., к.т.н., професор

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

затверджені наказом ЗНУ від “27” лютого 2025 року № 313-с

2. Строк подання студентом роботи 19.01.2026 р.

3. Вихідні дані до роботи: Мікроконтролер nRF52840 (Bluetooth 5.0 Long Range), мікроконтролерний модуль ESP32-S3-WROOM-1, інструментальний підсилювач ЕКГ AD8232, керамічна патч-антена SWLP.2450, лімієвий елемент живлення CR2032, стабілізатор напруги AMS1117-3.3, WEB-система на базі Laravel 10 та Vue.js.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1 Аналіз сучасних мікроелектронних компонентів для проведення спортивних змагань і розробки Web-систем. 2 Розробка мікроелектронної системи для проведення спортивних змагань. 3 Техніко-економічне обґрунтування. 4 Охорона праці та техногенна безпека

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

Л1 – Структурна схема системи; Л2– Алгоритм функціонування програмного забезпечення інтелектуального ілюзу; Л3 – Принципова електрична схема нагрудного датчика пульсу; Л4– Принципова електрична схема приймального ілюзу; Л5– Структурна схема інформаційної Web-системи; Л6- Алгоритм

функціонування інформаційної Web-системи; Л7– ER-діаграма бази даних інформаційної web-системи; Л8 – Схема веб-інтерфейсу інформаційної системи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
I	Ніконова З.А., проф. каф. ЕІСПЗ	27.02.25	23.03.25
II	Ніконова З.А., проф. каф. ЕІСПЗ	10.05.25	06.06.25
III	Ніконова З.А., проф. каф. ЕІСПЗ	13.10.25	03.11.25
IV	Ніконова З.А., проф. каф. ЕІСПЗ	10.11.25	26.12.25

6. Консультанти розділів кваліфікаційної роботи магістра

7. Дата видачі завдання 27.02.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи магістра	Строк виконання етапів кваліфікаційної роботи магістра	Примітка
1	<i>Аналіз матеріалу за темою кваліфікаційної роботи</i>	<i>23.03.25 -20.04.25</i>	
2	<i>Аналіз сучасних мікроелектронних технологій</i>	<i>21.04.25 -20.05.25</i>	
3	<i>Дослідження сучасних датчиків пульсу, мікроконтролерів, засобів для розробки web-систем</i>	<i>22.05.25 -26.06.25</i>	
4	<i>Розробка загальної структурної схеми системи</i>	<i>01.07-15.07.25</i>	
5	<i>Створення принципової електричної схеми датчику пульсу та приймального шлюзу системи</i>	<i>16.07.25-10.09.25</i>	
6	<i>Розробка програмного забезпечення web-системи</i>	<i>11.09.25 -1.11.25</i>	
7	<i>Розділ «Техніко-економічне обґрунтування»</i>	<i>2.11.25-15.11.25</i>	
8	<i>Розділ «Охорона праці та техногенна безпека»</i>	<i>16.11.25 -30.11.25</i>	
9	<i>Оформлення пояснювальної записки, виконання креслень</i>	<i>1.12.25 -04.01.26</i>	

Студент _____ Шарапов І.С.
(підпис) (прізвище та ініціали)

Керівник роботи (проєкту) _____ Ніконова З.А.
(підпис) (прізвище та ініціали)

Нормоконтроль пройдено

Нормоконтролер _____ Верьовкін Л. Л. (прізвище та ініціали)

РЕФЕРАТ

Дипломна робота містить 107 сторінок, 37 рисунків, 20 таблиць, 20 джерел літератури, додатків – 7.

Об'єкт дослідження – процес дистанційного моніторингу серцевої активності спортсменів у режимі реального часу за допомогою мікроелектронних пристроїв та веб-технологій.

Мета роботи – розробка мікроелектронної системи для безпроводного моніторингу частоти серцевих скорочень на спортивних змаганнях.

Завдання роботи – розробка архітектури системи, проєктування принципових електричних схем нагрудного датчика та інтелектуального шлюзу, розробка програмного забезпечення мікроконтролерів, створення WEB-системи для централізованого прийому й аналізу телеметрії.

Методика досліджень – моделювання програмно-апаратних алгоритмів та електричних схем у середовищах Wokwi та EasyEDA Pro; використання локального web-серверу для розробки і тестування web-системи.

Короткий виклад результатів досліджень – у роботі синтезовано систему, що передає дані про серцеві скорочення від датчику пульсу через інтелектуальний шлюз на базі ESP32-S3, який автоматично трансформує біометричні дані у JSON-формат та ретранслює їх через Wi-Fi на web-систему.

Прогнозні пропозиції – рекомендовано подальше розширення системи шляхом впровадження Mesh-топології для необмеженого покриття стадіонів та розроблення алгоритмів машинного навчання для автоматичної діагностики перевтоми спортсменів.

МІКРОКОНТРОЛЕР, ДАТЧИК ПУЛЬСУ, BLUETOOTH LONG RANGE, АЦП, ЕКГ, JSON, БАЗА ДАНИХ, WEB-СИСТЕМА

Кваліфікаційну роботу виконано на кафедрі електроніки, інформаційних систем та програмного забезпечення з 27.02.2025р. по 19.01.2026 р.

ЗМІСТ

ВСТУП.....	7
1. АНАЛІЗ СУЧАСНИХ МІКРОЕЛЕКТРОННИХ КОМПОНЕНТІВ ДЛЯ ПРОВЕДЕННЯ СПОРТИВНИХ ЗМАГАНЬ ТА РОЗРОБКИ WEB-СИСТЕМ.....	9
1.1 Датчики пульсу.....	10
1.2 Види датчиків пульсу	11
1.2.1 Нагрудні датчики пульсу	14
1.3 Мікроконтролери	20
1.3.1 Вибір мікроконтролеру для системи проведення спортивних змагань	21
1.4 Аналіз сучасних WEB-технологій.....	26
2. РОЗРОБКА МІКРОЕЛЕКТРОННОЇ СИСТЕМИ ДЛЯ ПРОВЕДЕННЯ СПОРТИВНИХ ЗМАГАНЬ.....	30
2.1 Розробка структурної схеми системи.....	30
2.2 Алгоритм функціонування програмного забезпечення інтелектуального шлюзу	32
2.3 Розробка датчику пульсу	35
2.3.1 Розробка вузлу аналогового інтерфейсу (AFE).....	36
2.3.2 Мікроконтролер nRF52840.....	37
2.3.3 Антена та блок живлення.....	39
2.3.4 Принципова електрична схема нагрудного датчика пульсу	40
2.3.5 Програмне забезпечення для датчику пульсу	42
2.4 Розробка приймального шлюзу	44
2.4.1 Мікроконтролер ESP32.....	44
2.4.2 Блок живлення для мікроконтролеру приймального шлюзу.....	45
2.4.3 Блок індикації.....	46
2.4.4 Принципова електрична схема приймального шлюзу	47
2.4.5 Розробка програмного забезпечення мікроконтролеру приймального шлюзу	49
2.5 Розробка WEB-системи	50
2.5.1 Проектування структури та архітектури інформаційної системи.....	51
2.5.2 Алгоритм роботи web-системи.....	52

2.5.3 Проектування бази даних інформаційної системи	54
2.5.4 Розробка програмного забезпечення інформаційної web-системи ...	59
3. ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ.....	69
4. ОХОРОНА ПРАЦІ ТА ТЕХНОГЕННА БЕЗПЕКА	77
4.1 Електробезпека при роботі з електроприладами та офісною оргтехнікою	77
4.2 Безпека при роботі з датчиками пульсу	79
4.3 Пожежна безпека та експлуатація автономних джерел живлення.....	81
ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ	82
ПЕРЕЛІК ПОСИЛАНЬ.....	84
ДОДАТОК А.....	87
ДОДАТОК Б.....	88
ДОДАТОК В.....	92
ДОДАТОК Г.....	97
ДОДАТОК Д.....	98
ДОДАТОК Е.....	100
ДОДАТОК Є.....	104

ВСТУП

У сучасному світі спорт давно перестав бути лише розвагою чи способом підтримання фізичної форми. Він є невід'ємною частиною суспільного життя, потужним інструментом для виховання здорового способу життя, формування характеру, розвитку особистісних якостей, зміцнення міжособистісних зв'язків та навіть національної ідентичності. Від масових забігів і шкільних змагань до Олімпійських ігор – спортивні змагання об'єднують мільйони людей по всьому світу, даруючи емоції, надихаючи на досягнення та демонструючи межі людських можливостей.

Важливою тенденцією останніх десятиліть є розширення кола учасників спортивних змагань. Сучасне суспільство дедалі більше усвідомлює необхідність інклюзії та доступності спорту для всіх верств населення. Це стосується не лише професійних спортсменів, а й дітей, які роблять свої перші кроки у спорті, осіб з інвалідністю, для яких спорт стає засобом реабілітації та соціалізації, аматорів, що прагнуть покращити свої результати, та ветеранів, які підтримують активний спосіб життя. Для ефективного та справедливого проведення змагань у таких різноманітних категоріях виникає потреба в адаптованих, точних та надійних системах фіксації результатів та моніторингу.

Стрімкий розвиток інформаційних та мікроелектронних технологій не оминув і спортивну галузь, внісши суттєві зміни у методи підготовки спортсменів, проведення змагань, аналіз результатів та взаємодію з аудиторією.

Одним з ключових аспектів актуальності даної теми є здатність мікроелектронних компонентів забезпечувати точність та об'єктивність у визначенні результатів. Мікроелектронні системи, такі як фотофінішні комплекси, електронні таймери, системи автоматичного вимірювання відстаней та висот, усувають ці недоліки, забезпечуючи миттєву та беззаперечну фіксацію досягнень. Це гарантує чесність та прозорість змагань, що є фундаментальним принципом спорту.

Окрім фіксації результатів, мікроелектронні пристрої відіграють важливу роль у моніторингу стану спортсменів. Застосування датчиків пульсу, акселерометрів, GPS-трекерів, сенсорів температури тіла та інших біометричних показників дозволяє в реальному часі відстежувати фізичний стан атлета під час тренувань та змагань. Це не тільки сприяє оптимізації тренувального процесу, допомагаючи тренерам адаптувати навантаження та стратегії, але й, що особливо важливо, підвищує безпеку спортсменів. Своєчасне виявлення перевтоми, аномальних показників чи потенційних проблем зі здоров'ям може запобігти травмам та іншим негативним наслідкам.

Інтеграція мікроелектронних технологій із засобами доступу до інформації через інтернет відкриває нові можливості для якісного перетворення системи проведення спортивних змагань. Розробка такої системи є актуальним завданням, яке відповідає сучасним вимогам до точності, об'єктивності, безпеки та доступності даних у спорті.

У даній роботі будуть розглянуті мікроелектронні компоненти інформаційної системи для збору інформації про стан спортсменів, а саме: датчик пульсу, мікроконтролер та WEB-сервер. Буде проведений аналіз існуючих мікроелектронних систем та технологій, що застосовуються цієї мети. Також у роботі обґрунтовується вибір апаратної бази (мікроконтролерів, сенсорів, модулів зв'язку) для реалізації компонентів системи.

1. АНАЛІЗ СУЧАСНИХ МІКРОЕЛЕКТРОННИХ КОМПОНЕНТІВ ДЛЯ ПРОВЕДЕННЯ СПОРТИВНИХ ЗМАГАНЬ ТА РОЗРОБКИ WEB-СИСТЕМ

Мікроелектроніка є фундаментом для створення пристроїв, які забезпечують необхідний рівень точності та контролю. Від високошвидкісних камер фотофінішу та лазерних систем вимірювання до мініатюрних сенсорів, вбудованих у спортивний одяг чи інвентар, – ці компоненти є основою для об'єктивізації спортивного процесу. Сучасні системи реєстрації результатів базуються на мікроконтролерах, які здатні швидко обробляти великі масиви даних, отриманих від різноманітних датчиків, і забезпечувати миттєву передачу інформації.

Одним із найбільш важливих напрямків застосування мікроелектроніки у спорті є моніторинг фізіологічних показників спортсменів. Можливість безперервного відстеження частоти серцевих скорочень (ЧСС), рівня кисню в крові, температури тіла, параметрів руху (швидкість, прискорення, подолана відстань) дозволяє тренерам та спортивним медикам здійснювати індивідуалізацію тренувальних навантажень, запобігати перевтомі та своєчасно виявляти ризики для здоров'я. Зокрема, використання нагрудних пульсометрів у поєднанні з портативними мікроконтролерними модулями стало стандартною практикою для забезпечення безпеки та оптимізації ефективності тренувань.

Зібрані мікроелектронними пристроями дані мають цінність лише за умови їх швидкої та доступної обробки й презентації. Тут на сцену виходять веб-системи. Необхідність у централізованому сховищі даних, можливості

аналізу в реальному часі, а також необхідність забезпечення доступу до інформації для всіх зацікавлених сторін (спортсменів, тренерів, організаторів, уболівальників) зумовлюють розробку потужних інформаційно-аналітичних платформ. Веб-технології, зокрема використання мов програмування та фреймворків, як-от PHP, JavaScript, а також різних систем керування базами даних, надають ідеальний інструментарій для створення масштабованих, гнучких та користувацьких інтерфейсів для доступу до спортивних даних.

Таким чином, розробка сучасної системи для проведення спортивних змагань є завданням системної інтеграції: об'єднання точних і надійних мікроелектронних компонентів для збору даних (наприклад, нагрудний пульсометр з мікроконтролером для попередньої обробки) із потужним і доступним веб-сервісом для передачі, зберігання та аналізу цієї інформації. У наступних підрозділах буде проведено детальний аналіз ключових технологічних компонентів, необхідних для реалізації такого проєкту.

1.1 Датчики пульсу

Пульсометр – це електронний пристрій, призначений для безперервного спостереження за частотою скорочень серця у реальному часі або для фіксації показників з метою подальшого аналізу. Цей тип датчиків знайшов широке застосування серед осіб, які активно займаються спортом або регулярними фізичними тренуваннями.

Історично перші зразки пульсометрів мали відносно просту конструкцію, що складалася з корпусу та двох електродів, які фіксувалися на грудній клітці. Їхнім основним призначенням був виключно моніторинг пульсу. Проте, у більш сучасних моделях пульсометрів, окрім основного датчика серцевого ритму, інтегровані додаткові функції, що підвищують їхню

функціональність та зручність використання. До таких функцій належать: годинник, секундомір, крокомір, календар та інші. Це розширення функціоналу відображає тенденцію до створення багатоцільових пристроїв для комплексного моніторингу фізичної активності [1].

1.2 Види датчиків пульсу

Існує низка класифікацій датчиків вимірювання частоти серцевих скорочень (ЧСС), що ґрунтуються на різних технічних критеріях, таких як метод вимірювання, тип передачі сигналу та спосіб підключення.

Основні методи вимірювання ЧСС поділяються на механічні, електричні та оптичні. Механічний метод реалізований у тонометрах, де фіксація ЧСС відбувається шляхом реєстрації змін тиску, зумовлених роботою серця. Електричний метод базується на реєстрації мікрострумів, що генеруються в процесі скорочення серцевих м'язів. Цей принцип широко використовується в обладнанні для електрокардіографії (ЕКГ) та в нагрудних датчиках вимірювання пульсу. Однак, суттєвим недоліком як механічних, так і електричних методів є необхідність забезпечення щільного фізичного контакту з тілом. Для механічного способу датчик має бути безпосередньо притиснутий до шкіри в місцях неглибокого залягання судин, тоді як електричні датчики вимагають розміщення якомога ближче до серця.

На противагу зазначеним методам, оптичний пульсометр, що набув поширення в портативній електроніці, функціонує на основі технології фотоплетизмографії. Даний датчик містить світлодіоди, які випромінюють світло, та фотодетектори, що реєструють рівень його відбиття. Принцип роботи заснований на тому, що в дермі людини розташована велика кількість капілярів, наповнених кров'ю. Під час систоли серця тиск підвищується, що

призводить до інтенсивнішого кровонаповнення судин і, відповідно, до більшого поглинання світла. Датчик фіксує ці зміни в поглинанні світла, і шляхом підрахунку кількості таких "сплесків" за одиницю часу (хвилину) визначається частота серцевого ритму [2].

Залежно від розташування датчики класифікуються на нагрудні, вбудовані та датчики пульсації крові. Нагрудний датчик вважається найбільш точним засобом визначення пульсу. Він фіксується на грудях за допомогою спеціального ремня та має автономне джерело живлення, що активується при реєстрації пульсу. Зібраний сигнал потім бездротовим способом передається на приймач. Перевагою цього типу датчиків є висока точність показань незалежно від погодних умов, проте недоліком може бути дискомфорт при тривалому використанні.

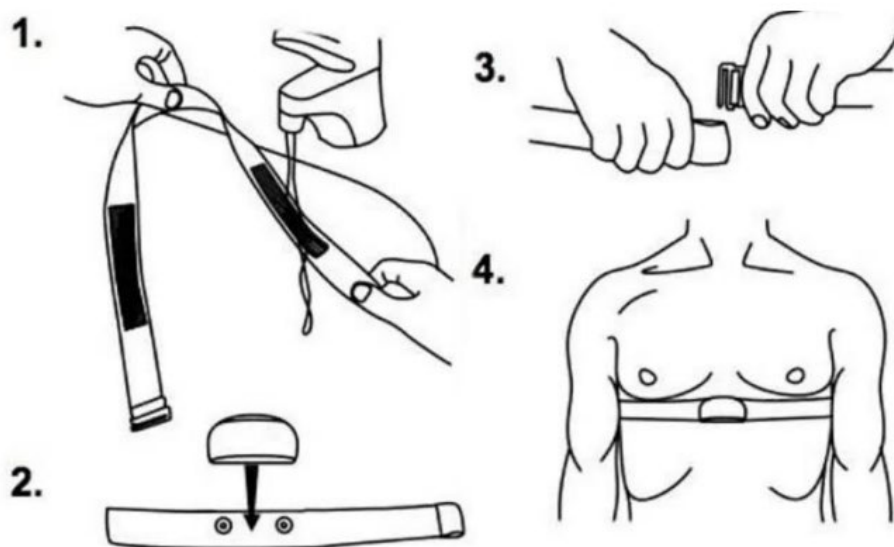


Рисунок 1.1 – Кріплення нагрудного датчика

Вбудовані датчики, що реалізуються у сучасних пульсометрах без нагрудного ремня, визначають пульс за допомогою двох електродів, розташованих на корпусі пристрою та прикладаються до тіла на короткий час. Ці пристрої, представлені у форм-факторах фітнес-браслетів, годинників або пульсометрів-кілець, популярні завдяки зручності та простоті експлуатації, хоча їхня точність може поступатися нагрудним датчикам. Основними

перевагами є легкість використання, оскільки датчик інтегрований у пристрій для безпосереднього контакту з тілом. Однак, на їх точність суттєво впливають низькі температури або нечітка фіксація на тілі.

Датчики пульсації крові, що кріпляться на мочку вуха або палець, визначають ритм серця за пульсацією крові у тканинах. Інформація відображається на дисплеї компактного пристрою. Перевагами датчиків на мочку вуха є легкість та компактність, але вони можуть викликати дискомфорт (перетискання або спадання), а також схильні до неточних показань при холодній погоді. Датчики на палець, що кріпляться у вигляді прищіпки або рукавиці, після звикання практично не відчуються на руці, але також можуть надавати неточні показання за низьких температур. Окремим типом є навушники-пульсометри, що мають вбудований монітор серцебиття, дозволяючи поєднувати прослуховування музики з контролем пульсу, хоча їхній недолік полягає у потенційному відволіканні.

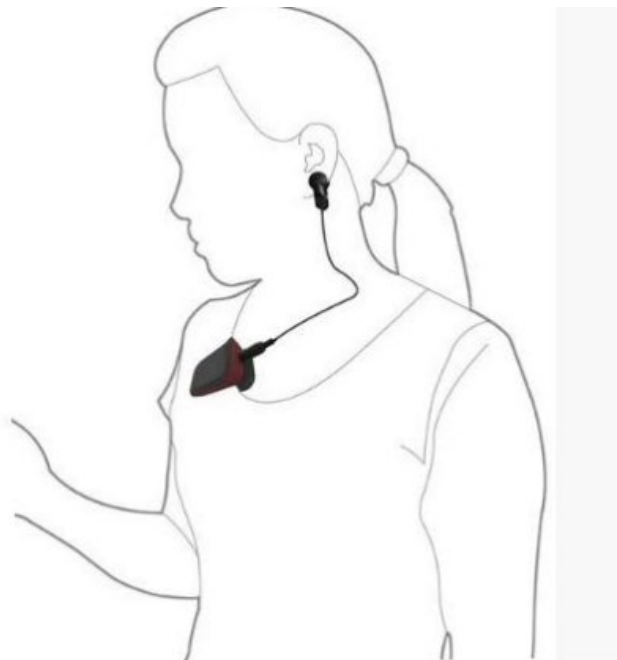


Рисунок 1.2 – Датчик пульсації крові з кріпленням на мочку вуха

За типом передачі сигналу датчики класифікуються на аналогові та цифрові. Аналогові датчики підключаються до пристроїв, таких як тренажери

або смартфони, і передають некодований сигнал. Це може призводити до перехресного зчитування даних від сусідніх аналогових пульсометрів, а також до збоїв у роботі поблизу джерел електромагнітних перешкод, наприклад, високовольтних ліній. Цифрові датчики, натомість, забезпечують більш точні показання та є стійкішими до зовнішніх перешкод.

За способом підключення датчики поділяються на дротові та бездротові. Дротові датчики з'єднані з основним приладом за допомогою кабелю, що забезпечує надійність з'єднання та високу стійкість до перешкод. Бездротові датчики передають дані по радіоканалу, як в аналоговому, так і в цифровому форматі. Недоліками бездротових систем є необхідність заміни елементів живлення та потенційна вразливість до порушень передачі сигналу в умовах радіоперешкод.

1.2.1 Нагрудні датчики пульсу

Для мікроелектронної системи, призначеної для використання на спортивних змаганнях, вибір нагрудних датчиків пульсу є найбільш обґрунтованим рішенням, незважаючи на їхні потенційні недоліки у повсякденному використанні. Основним аргументом на користь нагрудних датчиків є їхня неперевершена точність та надійність отримання даних. У змагальних умовах, де кожна секунда та показник мають вирішальне значення для аналізу ефективності та безпеки спортсмена, точність вимірювань, забезпечена прямим електричним контактом біля серця, стає пріоритетом. На відміну від оптичних датчиків, точність яких може знижуватися під впливом зовнішніх факторів (рух, піт, холод, некоректна фіксація на тілі), нагрудні датчики мінімально схильні до таких перебоїв, забезпечуючи стабільний та

достовірний сигнал навіть за інтенсивних фізичних навантажень та в складних погодних умовах.

Для спортсменів, зокрема бігунів, використання нагрудного датчика пульсу часто є більш зручним та функціональним рішенням порівняно з тими, що вдягаються на палець, навушними або інтегрованими в інші елементи екіпіровки датчиками. Під час інтенсивного бігу або інших динамічних вправ важливо мінімізувати фактори, що можуть відволікати або створювати дискомфорт. Нагрудний датчик, закріплений на грудях, як правило, надійно фіксується і не схильний до зсувів, що можуть виникнути при рухах рук (як у випадку з тими, що фіксуються на палець) або голови (для навушників). Це забезпечує стабільність контакту та, відповідно, безперервність і точність вимірювань без необхідності постійного коригування положення датчика. Хоча вдягаємі на палець та навушники можуть здаватися зручнішими у стані спокою, у динаміці бігу вони схильні до перетискання, спадання, або ж можуть заважати зосередитися на тренуванні, що є неприпустимим під час спортивних змагань або тривалих інтенсивних пробіжок. Саме тому нагрудний датчик – найбільш правильний вибір для спортивних змагань.

Розглянемо конкретні приклади нагрудних датчиків, що присутні на ринку. Вимоги, яким повинні відповідати такі датчики:

- Висока точність вимірювання ЧСС: здатність надавати максимально точні дані про частоту серцевих скорочень в умовах інтенсивних фізичних навантажень та швидких змін ритму.
- Надійний бездротовий інтерфейс передачі даних: підтримка стандартних протоколів: Сумісність з широко поширеними бездротовими протоколами (наприклад, ANT+, Bluetooth Low Energy (BLE) чи Bluetooth 5.0), що забезпечує легку інтеграцію з різноманітними приймачами та мікроконтролерами.
- Стійкість до перешкод: Ефективна робота в умовах високої щільності радіосигналів (інші датчики, Wi-Fi, стільниковий зв'язок) на змаганнях.

- Достатній радіус дії: Забезпечення стабільної передачі даних на відстані, достатній для зв'язку між спортсменом та приймачем системи (наприклад, до 10-20, в ідеалі до 100-200 метрів).
- Сумісність з мікроконтролерами: наявність чітко документованого протоколу зв'язку та, бажано, готових бібліотек або прикладів коду для інтеграції з популярними мікроконтролерними платформами (наприклад, Arduino, ESP32, Raspberry Pi).
- Автономне джерело живлення та тривалий час роботи:
- Фізична міцність та ергономічність:
- Вологостійкість/Водонепроникність: захист від поту та опадів.
- Стійкість до ударів та вібрацій: Конструкція, здатна витримувати типові навантаження під час спортивних змагань.
- Комфортне кріплення: Надійний та гіпоалергенний ремінь, що забезпечує щільне прилягання до шкіри без натирання та дискомфорту.
- Температурна стійкість: Здатність функціонувати в широкому діапазоні температур, характерних для зовнішніх спортивних подій.

Існує декілька датчиків, що відповідають цим вимогам. Розглянемо найбільш підходящі з них.

Polar H10 – сучасний датчик пульсу. Цей нагрудний пульсометр вирізняється розширеними можливостями підключення, підтримуючи універсальні бездротові протоколи Bluetooth Smart LE (з можливістю одночасного з'єднання до двох пристроїв) та ANT+, що забезпечує високу сумісність з більшістю спортивних гаджетів та мікроелектронних систем. Додатковою перевагою є наявність протоколу GymLink 5 kHz, що дозволяє використовувати датчик зі студійним фітнес-обладнанням та навіть здійснювати зчитування пульсу під водою. Пристрій демонструє виняткову точність вимірювання, що відповідає рівню електрокардіограми (ECG), з похибкою менше 1%, роблячи його одним із найнадійніших у своєму класі. Живлення здійснюється від стандартного елемента CR2025, що забезпечує 400 годин автономної роботи. [3].



Рисунок 1.3 – Нагрудний пульсометр Polar H10 N

Garmin HRM-Pro Plus – нагрудний пульсометр, що забезпечує гнучке підключення завдяки підтримці обох основних бездротових протоколів: ANT+® та Bluetooth Low Energy, що гарантує широку сумісність з екосистемами Garmin Connect, Tascx, а також численними сторонніми спортивними додатками. Пристрій вирізняється розширеними можливостями моніторингу, надаючи дані динаміки бігу, такі як вертикальні коливання, час контакту з землею та довжина кроку, що є критично важливим для глибокого аналізу та оптимізації техніки бігу. Вбудована пам'ять дозволяє зберігати дані про час тренування та частоту серцевих скорочень (HR) навіть у випадку тимчасової втрати з'єднання, з подальшою автоматичною синхронізацією при відновленні зв'язку. Живлення здійснюється від батареї CR2032, що забезпечує тривалий термін служби – до 1 року в активному режимі [4].



Рисунок 1.4 – Garmin HRM-Pro Plus

Wahoo TICKR X – багатофункціональний пристрій, що пропонує універсальні можливості підключення завдяки підтримці двосмугових

протоколів ANT+ та Bluetooth LE, дозволяючи одночасно підключатися до трьох пристроїв через Bluetooth. Пристрій оснащений вбудованою пам'яттю, здатною зберігати до 50 годин даних про частоту серцевих скорочень (HR), що забезпечує гнучкість тренувань без постійного зв'язку з приймачем. Живлення здійснюється від батареї CR2032, яка гарантує понад 500 годин безперервної роботи [5].



Рисунок 1.5 – Wahoo TICKR X

Тепер зробимо порівняльний аналіз даних моделей (Таблиця 1.1):

Таблиця 1.1 – Порівняльний аналіз моделей пульсометрів

Показник	Polar H10	Garmin HRM-Pro Plus	Wahoo TICKR X
BLE + ANT+	✓ + GymLink	✓ + ANT+ (Garmin)	✓ + ✓
Локальна пам'ять	40 годин	так (плаваюче)	50 годин
Точність	ECG-рівень <1 %	дуже висока	клінічно добра
Життєдіяльність батареї	~400 годин	~1 рік	~500 годин
Вартість (орієнтир)	4000-5000 грн	8500-10000 грн	4500-5500
Наявність SDK / інтеграції	відкритий, широке співтовариство	Garmin-SDK (закритий)	Wahoo API + стріми

Незважаючи на високу точність та надійність комерційних нагрудних пульсометрів (Polar H10, Garmin HRM-Pro Plus, Wahoo TICKR X), які були розглянуті вище, їхня інтеграція в єдину, централізовану інформаційну систему моніторингу змагань стикається з низкою критичних обмежень, що вимагає розробки власного мікроелектронного модуля.

По-перше, основоположною перешкодою є закрита архітектура та обмежена можливість перепрограмування вбудованих мікроконтролерів комерційних пристроїв. Ці продукти розроблені як кінцеві пристрої з фіксованим функціоналом і не передбачають втручання в їхню прошивку для зміни протоколів передачі даних або додавання специфічних функцій. Розробка власного модуля дозволяє повністю контролювати алгоритм збору, фільтрації та відправки даних, забезпечуючи повну сумісність між нагрудним датчиком та центральним мікроконтролером системи.

По-друге, комерційні рішення переважно використовують стандартний протокол Bluetooth Low Energy (BLE) з обмеженим радіусом дії, що відповідає їхньому типовому сценарію використання (зв'язок зі смартфоном або годинником спортсмена). Для масштабованої системи моніторингу спортивних змагань критично важливо забезпечити передачу даних на значні відстані до одного або кількох стаціонарних приймальних пунктів. Це вимагає використання технології Bluetooth 5.0 Long Range (Далекий радіус дії), яка, хоча й підтримується сучасними чипами (наприклад, ESP32), не використовується у стандартних комерційних пульсометрах. Розробка власного модуля дозволяє інтегрувати цю технологію, суттєво розширюючи робочу зону системи.

По-третє, важливим фактором є економічна доцільність та можливість вибору компонентної бази. Вартість придбання значної кількості комерційних пульсометрів для забезпечення змагань (наприклад, 50-100 одиниць) є суттєвою. Створення власного пристрою "з нуля" дозволяє знизити собівартість одиниці. Додатково, це відкриває можливість використання вітчизняної (української) елементної бази там, де це можливо (наприклад, у

виробництві друкованих плат або корпусів), що не лише зменшує логістичні витрати, але й сприяє розвитку національного ринку мікроелектроніки. Таким чином, розробка власного мікроелектронного модуля є необхідною умовою для досягнення цілей проекту щодо масштабованості, контрольованості та економічної ефективності.

1.3 Мікроконтролери

Мікроконтролер – мікросхема, призначена для управління електронними пристроями. Типовий мікроконтролер поєднує на одному кристалі функції процесора і периферійних пристроїв, містить оперативну та постійну пам'яті. Такі схеми можуть виконувати задачі регулювання та управління електротехнічними, електроенергетичними та іншими пристроями, а також технологічними процесами виробництва, досліджень, випробувань [6]. Однокристальні мікроконтролери – це пристрої, які конструктивно виконані в одному корпусі і містять у собі всі складові мікропроцесорної системи: процесор, пам'ять даних та програм, інтерфейси. Високоінтегровані мікроконтролери можуть мати аналого-цифровий перетворювач, цифро-аналоговий, а також температурні сенсори, порти, таймери та ін. [7].

При модульному принципі побудови всі мікроконтролери одного сімейства містять процесорне ядро (однакове для всіх мікроконтролерів даного сімейства), і змінний функціональний блок, що відрізняє мікроконтролери різних моделей. Таке процесорне ядро містить у собі:

- центральний процесор;
- внутрішню контролерну магістраль, яка складається з шин адреси, даних і управління;
- схему синхронізації мікроконтролеру;

– схему управління режимами роботи мікроконтролера.

Сучасний мікроконтролер є складною цифровою системою, що розміщується на кристалі (рис. 1.6).



Рисунок 1.6 – Архітектура мікроконтролера

До складу мікроконтролеру входять:

- один процесор (8-,16– чи 32-х розрядний)
- внутрішня пам'ять програми та даних
- набір інтерфейсних і периферійних пристроїв. Зокрема: портів вводу/виводу, таймерів, аналого-цифрових перетворювачів (АЦП), цифро-аналогових перетворювачів (ЦАП) та інших [8].

1.3.1 Вибір мікроконтролеру для системи проведення спортивних змагань

Для реалізації безпроводного шлюзу, що забезпечує ретрансляцію даних від біометричних датчиків до WEB-системи, було проведено порівняльний аналіз трьох актуальних мікроконтролерних платформ: Arduino Nano 33 IoT, Raspberry Pi Pico W та ESP32 (WROOM-32).

Мікроконтролерна плата Arduino Nano 33 IoT позиціонується як економічно ефективне рішення для швидкої інтеграції пристроїв у домен Інтернету речей (IoT), що робить її придатною для розробки маломасштабних сенсорних мереж.

Ключовою апаратною основою плати є 32-бітний мікропроцесор Arm® Cortex®-M0 SAMD21, який вирізняється низьким енергоспоживанням, що критично для автономних систем моніторингу, як у нашому проєкті. Забезпечення бездротового зв'язку реалізовано за допомогою інтегрованого модуля u-blox NINA-W10, що підтримує роботу у діапазоні 2.4 ГГц та забезпечує функціонал Wi-Fi та Bluetooth® Low Energy (BLE). Важливою архітектурною перевагою NINA-W10 є його здатність функціонувати як клієнтський, так і хост-пристрій для обох протоколів, що є унікальною властивістю для мікроконтролерних платформ цього класу. Ця двосторонність забезпечує гнучкість у взаємодії з нагрудним датчиком пульсу (як центральний пристрій) та передачі даних на сервер (через Wi-Fi або як периферійний пристрій BLE). [9]

Незважаючи на сучасну архітектуру, об'єм оперативної пам'яті (RAM) становить лише 32 КБ, а тактова частота – 48 МГц. Для задач, що потребують одночасної підтримки важкого стеку протоколів Bluetooth та HTTP-запитів до веб-сервера, такий об'єм пам'яті є критично малим, що може призвести до нестабільної роботи системи (Stack Overflow) при обробці JSON-пакетів. До того ж, радіомодуль NINA-W102 базується на застарілій архітектурі ESP32-D0WD, яка підтримує лише Bluetooth версії 4.2. Це обмежує радіус дії системи 15–20 метрами в умовах прямої видимості.

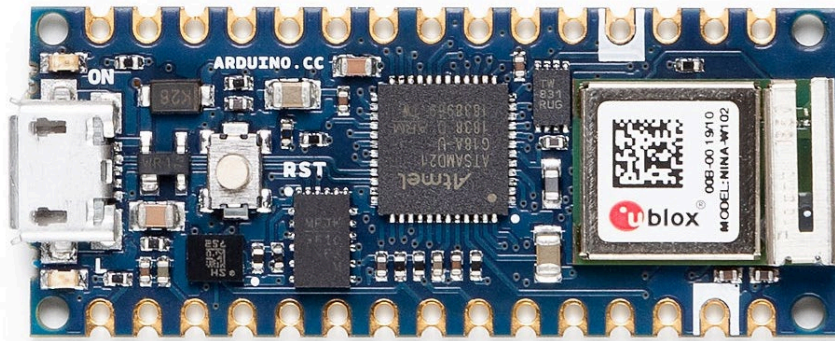


Рисунок 1.7 – Мікроконтролерна плата Arduino Nano 33 IoT

Плата Raspberry Pi Pico W є вдосконаленою версією мікроконтролера Raspberry Pi Pico, спеціально розробленою для інтеграції у проєкти Інтернету речей (IoT) завдяки вбудованому бездротовому зв'язку.

Ключовою апаратною основою Pico W є мікроконтролерний чип RP2040, розроблений компанією Raspberry Pi. Цей чип має двоядерну архітектуру, побудовану на базі процесорних ядер Arm Cortex-M0+, що працюють на тактовій частоті до 133 МГц. Чип RP2040 також оснащений 264 КБ SRAM та підтримує до 16 МБ зовнішньої флеш-пам'яті QSPI.

Для реалізації бездротового зв'язку на Pico W інтегровано модуль Infineon CYW43439, який забезпечує підтримку Wi-Fi у діапазоні 2.4 ГГц. Хоча апаратне забезпечення CYW43439 має можливість підтримувати Bluetooth Low Energy (BLE) та Bluetooth 5.2, на момент випуску плати підтримка BLE через офіційне програмне забезпечення не була реалізована, що значно ускладнює реалізацію проєкту [10].



Рисунок 1.8 – Мікроконтролерна плата Raspberry Pi Pico W

Мікроконтролерний модуль ESP32-S3 (WROOM-1) (розробник Espressif Systems) є одним із найбільш популярних рішень у сфері Інтернету речей (IoT), завдяки поєднанню високої продуктивності, інтегрованого бездротового зв'язку та економічної ефективності. Цей модуль є гарним кандидатом для розробки апаратного модуля мікроелектронної інформаційної системи, оскільки він задовольняє ключові вимоги проєкту: дводіапазонний бездротовий зв'язок та достатня обчислювальна потужність.

Обчислювальна архітектура ESP32 побудована на базі двоядерного 32-бітного мікропроцесора Tensilica Xtensa LX7, що може працювати на тактовій частоті до 240 МГц. Наявність двох незалежних ядер дозволяє ефективно розподіляти завдання: одне ядро може бути задіяне для виконання високопріоритетних завдань, таких як збір та обробка даних пульсу, а інше – для управління стеком протоколів бездротового зв'язку (Wi-Fi та Bluetooth). Модуль також має достатні обсяги вбудованої пам'яті: до 512 КБ SRAM та 4-16 МБ Flash-пам'яті (залежно від версії).

Ключовою перевагою ESP32 є повноцінна інтеграція дводіапазонного бездротового зв'язку (Dual-Mode Wireless). Модуль підтримує Wi-Fi 802.11 b/g/n (2.4 ГГц) для високошвидкісної передачі даних на сервер, а також має апаратну підтримку режиму Coded PHY стандарту Bluetooth 5.0 Long Range. Це дозволяє збільшити радіус прийому сигналу від нагрудного датчика до 150–250 метрів за рахунок використання методів прямого виправлення помилок (FEC). Підтримка цього стандарту є критично важливою для проєкту, оскільки вона дозволяє безпосередньо взаємодіяти з нагрудним датчиком пульсу, використовуючи стандартний профіль Heart Rate Profile (HRP) або власний протокол [11].

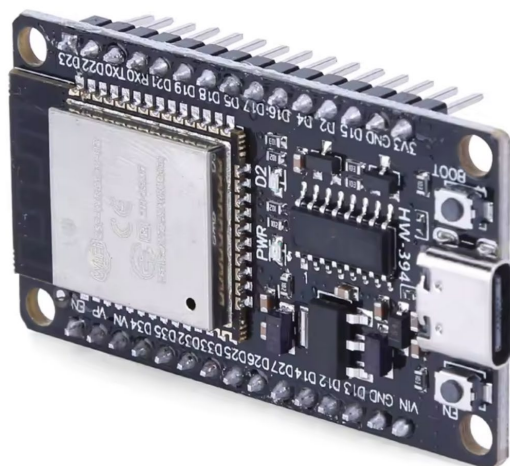


Рисунок 1.9 – Мікроконтролерна плата ESP32-WROOM-32

Проведемо порівняння цих мікроконтролерів (таблиця 1.2):

Таблиця 1.2 – Порівняння варіантів мікроконтролерів для системи проведення спортивних змагань

Параметр	Arduino Nano 33 IoT	Raspberry Pi Pico W	ESP32-S3 (WROOM-1)
Процесор	SAMD21 (Cortex-M0+)	RP2040 (Cortex-M0+)	Xtensa LX7
Кількість ядер	1	2	2
Тактова частота	48 МГц	133 МГц	240 МГц
SRAM	32 КБ	264 КБ	512 КБ + 2 МБ (PSRAM)
Flash-пам'ять	256 КБ	2 МБ	4 МБ
Bluetooth стек	Апаратний (u-blox)	Програмний	Апаратний (інтегрований)
Версія Bluetooth	4.2 (Legacy)	5.2	5.0 (Long Range)
Режим Coded PHY	Відсутній	Обмежений	Апаратна підтримка

Для реалізації безпроводного шлюзу було обрано мікроконтролер ESP32 S3. Порівняно з платформою Arduino Uno (яка потребує додаткових модулів) та Raspberry Pi Pico (яка має складнішу архітектуру стеку Bluetooth), ESP32 пропонує інтегроване рішення «System-on-Chip» з двоядерною архітектурою. Це дозволяє розділити критичні в часі завдання зняття показників пульсу та ресурсоемні завдання передачі даних по протоколу HTTP, забезпечуючи високу надійність системи.

1.4 Аналіз сучасних WEB-технологій

Майже безальтернативною мовою розмітки, що використовується в сучасній Web-розробці, є HTML. Це стандартизована мова розмітки документів у Всесвітній павутині [12]. Більшість веб-сторінок містять опис розмітки на мові HTML (або XHTML). Мова HTML інтерпретується браузерами; отриманий в результаті інтерпретації форматований текст відображається на екрані монітора комп'ютера або мобільного пристрою.

За візуальне представлення та стилізацію веб-інтерфейсу відповідає мова CSS (Cascading Style Sheets). Поточний стандарт, відомий як CSS3, є модульним, що означає його розвиток окремими специфікаціями (наприклад, Flexbox, Grid Layout, Transitions, Animations) під егідою консорціуму W3C (World Wide Web Consortium) та WHATWG [13].

Що стосується мови для програмування клієнтської частини, то з помітним відривом лідирує Javascript. Інші технології, такі як Flash, Java-аплети та інші у сучасній веб-розробці вже майже не використовуються.

JavaScript – мультипарадигменна мова програмування. Підтримує об'єктно-орієнтований, імперативний і функціональний стилі. Є реалізацією мови ECMAScript (стандарт ECMA-262 [14]). Найбільш широке застосування

знаходить в браузерях як мова сценаріїв для додання інтерактивності веб-сторінок. Основні архітектурні риси: динамічна типізація, слабка типізація, автоматичне керування пам'яттю, прототипне програмування, функції як об'єкти першого класу.

Для забезпечення високого рівня абстракції і досягнення прийняттого ступеня кросбраузерності при розробці веб-додатків використовуються бібліотеки та фреймворки JavaScript. Вони являють собою набір багаторазово використовуваних об'єктів і функцій. Серед відомих JavaScript фреймворків можна відзначити React, Angular, Vue.js. Для розробки проєкту було обрано фреймворк Vue.js. Vue.js є прогресивним фреймворком, який ідеально підходить для розробки клієнтської частини системи завдяки плавній інтеграції, високій продуктивності реактивної двосторонньої прив'язки даних (Two-Way Data Binding) та гнучкості (завдяки компонентам), що забезпечує швидке створення складних, але водночас зрозумілих і підтримуваних користувацьких інтерфейсів.

Основними мовами програмування для розробки серверної (back-end) частини веб-системи є PHP, Java, C#, Python та інші. Для розробки веб-проєкту була вибрана мова PHP. Це – скриптова мова загального призначення, інтенсивно застосовується для розробки веб-додатків. В даний час підтримується переважною більшістю хостинг-провайдерів і є одним з лідерів серед мов, що застосовуються для створення динамічних веб-сайтів. Ця мова має широке співтовариство з різних частин планети, що дозволяє швидко знаходити розповсюджені проблеми та вирішувати їх. Велика кількість форумів та різних обговорень дає можливість розраховувати на підтримку інших розробників в разі виникнення програмних питань, які не можна вирішити самотійно.

Популярність в області побудови веб-сайтів визначається наявністю великого набору вбудованих засобів для розробки веб-додатків. Основні з них:

- автоматичне вилучення POST і GET-параметрів, а також змінних оточення веб-сервера в визначені наперед масиви;

- взаємодія з великою кількістю різних систем управління базами даних;
- автоматизована відправка HTTP-заголовків;
- робота з HTTP-авторизацією;
- робота з cookies і сесіями;
- робота з локальними і віддаленими файлами, сокетом;
- обробка файлів, що завантажуються на сервер.

В даний час PHP використовується сотнями тисяч розробників. Для проєкту було обрано фреймворк Laravel. Він ґрунтується на архітектурі Model-View-Controller (MVC), що забезпечує чіткий поділ логіки, високу організованість та підтримуваність коду, критичну для складних інформаційних систем. Застосування Laravel дозволяє значно прискорити процес розробки завдяки вбудованим компонентам (Eloquent ORM для ефективної роботи з базою даних, системи маршрутизації та автентифікації), а також полегшує створення надійного API для прийому даних у реальному часі від мікроконтролерів (наприклад, через WebSockets) та забезпечення безпечної взаємодії з клієнтською частиною на Vue.js

У світі сучасної веб-розробки використовуються різні реляційні системи управління БД. Серед лідерів – MySQL, PostgreSQL, Oracle, Microsoft SQL Server та інші. В якості основної системи керування базами даних (СКБД) для серверної частини системи обрано MariaDB. Це рішення є форком (відгалуженням) популярної СКБД MySQL, розробленим та підтримуваним спільнотою, що гарантує його відкритість та стійкий розвиток. Ключові переваги MariaDB перед традиційною MySQL (особливо у контексті імплементації, що підтримується Oracle) полягають у підвищеній продуктивності, зокрема при обробці запитів з високим навантаженням, та розширених можливостях оптимізації завдяки впровадженню нових механізмів зберігання. Це критично важливо для системи, яка має обробляти та зберігати великі обсяги часових рядів даних пульсу в реальному часі.

В результаті аналізу сучасних мікроелектронних компонентів для проведення спортивних змагань ми можемо сформулювати мету та завдання кваліфікаційної роботи.

Мета роботи – розробка мікроелектронної системи для безпроводного моніторингу частоти серцевих скорочень на спортивних змаганнях.

Завдання роботи – розробка архітектури системи, проєктування принципових електричних схем нагрудного датчика та інтелектуального шлюзу, розробка програмного забезпечення мікроконтролерів, створення WEB-системи для централізованого прийому й аналізу телеметрії.

2. РОЗРОБКА МІКРОЕЛЕКТРОННОЇ СИСТЕМИ ДЛЯ ПРОВЕДЕННЯ СПОРТИВНИХ ЗМАГАНЬ

2.1 Розробка структурної схеми системи

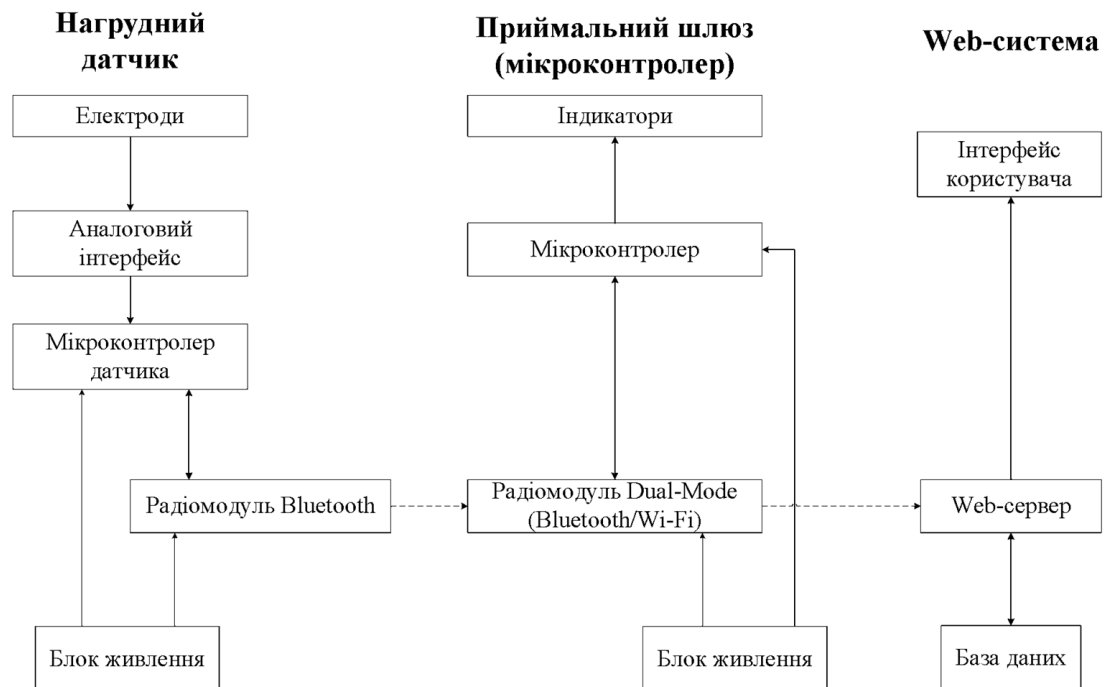


Рисунок 2.1 – Структурна схема системи

Представлена на рисунку 2.1 структурна схема відображає архітектуру апаратно-програмного комплексу для дистанційного моніторингу серцевої активності. Робота системи базується на принципі багатоступеневої передачі та обробки сигналу. Перший вузол – Нагрудний датчик – виконує роль первинного перетворювача. Біопотенціали серця через електроди надходять до аналогового інтерфейсу (AFE), де здійснюється їх підсилення та фільтрація від шумів. Мікроконтролер датчика оцифровує отриманий сигнал, вираховує

частоту серцевих скорочень та через радіомодуль Bluetooth 5.0 Long Range транслює дані в ефір. Блок живлення забезпечує автономність датчика, використовуючи енергоефективні режими роботи компонентів.

Генератор імпульсів призначений для узгодження роботи всіх вузлів пристрою, що розробляється (рис. 2.10), і забезпечує широку зміну частоти імпульсів (близько 50000 разів). Імпульсний генератор зібраний на елементах у вигляді мікросхеми K155ЛА3. Він забезпечує дослідження перехідних процесів мікроелектронної системи та визначення її параметрів. Перехідні процеси в ній відображають її динамічні властивості з заданим 40% –вим перерегулюванням на виходах каналів об'єкта управління та визначення величини періоду їх квантування у часі, а от же являються джерелом прямих показників якості управління (u_c, u_d, t_{pk}).

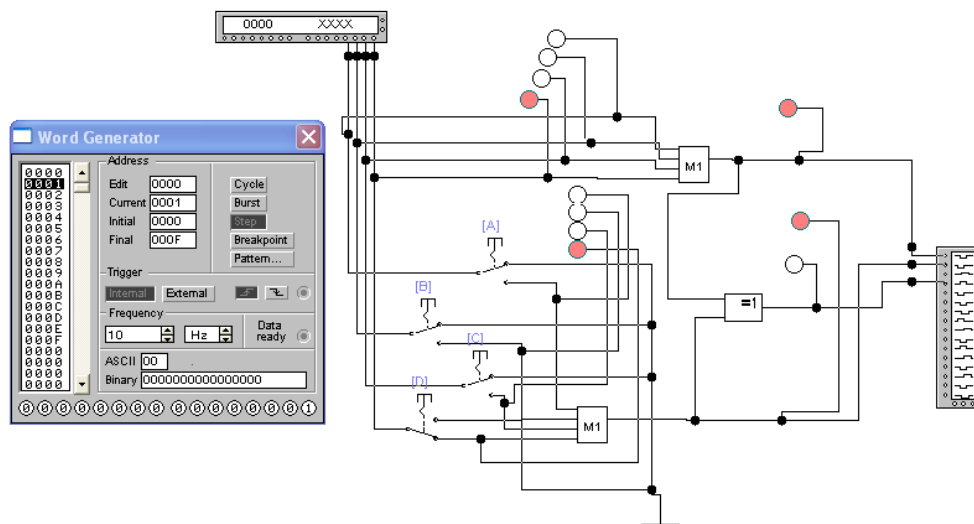


Рисунок 2.2 – Експериментальна схема генератора імпульсів

Другий вузол – Приймальний шлюз – виступає інтелектуальним посередником. Радіомодуль Dual-Mode, інтегрований у мікроконтролер ESP32, перехоплює Bluetooth-пакети від датчика. Центральний процесор шлюзу виконує дешифрацію даних, керує локальною індикацією (світлодіодним або рідкокристалічним дисплеєм для візуального контролю на місці) та ініціює TCP/IP з'єднання. Використання подвійного стеку протоколів

дозволяє одночасно підтримувати зв'язок із сенсором та передавати сформовані JSON-об'єкти через Wi-Fi канал до глобальної мережі.

Завершальним етапом є Web-система. Дані через HTTP-запити потрапляють на Web-сервер, де за допомогою серверної логіки (на базі PHP) відбувається їх верифікація та збереження у базі даних MariaDB. Кінцевий користувач або адміністратор змагань отримує доступ до результатів через інтерфейс користувача (HTML/JS сторінка), яка в реальному часі відображає динаміку пульсу. Схема демонструє чітку ієрархію інтерфейсів: аналоговий сигнал на вході, Bluetooth-протокол на середньому рівні та стандартні веб-протоколи на виході, що забезпечує масштабованість та надійність всієї системи.

2.2 Алгоритм функціонування програмного забезпечення інтелектуального шлюзу

На основі розробленої структурної схеми було спроектовано алгоритм роботи програмного забезпечення центрального мікроконтролера ESP32, який виконує роль шлюзу між персональною мережею датчика (PAN) та глобальною мережею (WAN). Алгоритм побудований за принципом лінійної ініціалізації з переходом у нескінченний цикл обробки переривань та подій. Алгоритм представлений на рисунку 2.3.

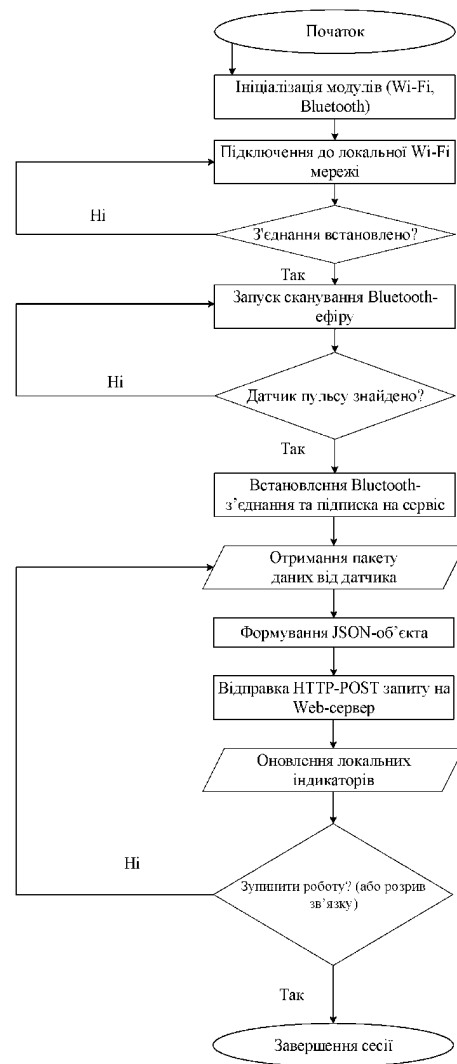


Рисунок 2.3 – Алгоритм функціонування програмного забезпечення інтелектуального шлюзу

Робота системи починається з блоку ініціалізації периферійних модулів. На цьому етапі відбувається конфігурація внутрішніх реєстрів мікроконтролера, налаштування шини I2C/SPI для роботи з індикаторами та активація стеку радіочастотних протоколів.

Першою критичною дією є встановлення Wi-Fi з'єднання. Мікроконтролер намагається підключитися до заданої точки доступу з використанням протоколу WPA2. У випадку невдачі алгоритм передбачає циклічне повторення запитів, що забезпечує відмовостійкість системи у разі

тимчасової відсутності мережі. Після отримання локальної IP-адреси пристрій переходить до сканування Bluetooth-ефіру. Процес сканування полягає у фільтрації пакетів від навколишніх Bluetooth-пристроїв з метою пошуку специфічного датчику пульсу.

Після ідентифікації потрібного датчика відбувається процедура Handshake (рукостискання) та встановлення з'єднання. Алгоритм виконує «підписку» (Subscription) на оновлення характеристик датчика. Це дозволяє реалізувати подієво-орієнтовану модель: мікроконтролер не витрачає ресурси на постійні запити, а переходить у режим очікування сповіщень, які надсилає датчик у момент реєстрації кожного удару серця.

Центральна частина алгоритму виконується циклічно і включає наступні кроки:

1. Отримання даних: при надходженні Bluetooth-пакету спрацьовує функція зворотного виклику (callback), яка витягує значення частоти серцевих скорочень (BPM) у байтовому форматі.
2. Формування пакету: отримане числове значення інкапсулюється у формат JSON. Вибір цього формату обумовлений його легкістю та повною сумісністю з розробленою WEB-системою на базі PHP. Окрім пульсу, у пакет додається ідентифікатор пристрою (ID) та статусний код.
3. Передача даних: сформований пакет передається на Web-сервер за допомогою HTTP POST-запиту. Використання POST-методу замість GET забезпечує кращу цілісність даних та відповідає стандартам розробки сучасних API.
4. Візуалізація: паралельно з мережевою активністю алгоритм оновлює стан локальних індикаторів. Це необхідно для оперативного контролю працездатності системи безпосередньо на місці проведення змагань.

Алгоритм передбачає постійну перевірку статусу з'єднань. У разі розриву зв'язку з датчиком або Wi-Fi роутером, система автоматично

повертається на відповідний етап перепідключення. Завершення роботи відбувається при повному знеструмленні пристрою або при надходженні команди на зупинку сесії моніторингу, після чого мікроконтролер коректно закриває мережеві сокети.

2.3 Розробка датчику пульсу

Для побудови інтелектуального нагрудного датчика пульсу нам необхідні наступні компоненти: система сухих текстильних або гумових електродів для зняття біопотенціалів, аналоговий фронтенд (AFE) на базі спеціалізованої мікросхеми AD8232 для підсилення та фільтрації ЕКГ-сигналу, центральний мікроконтролер (SoC) nRF52840 з інтегрованим радіомодулем Bluetooth 5.0 Long Range для обробки та трансляції даних, прецизійний кварцовий резонатор на 32 МГц для стабілізації радіоканалу, блок автономного живлення на основі літієвої батареї CR2032, а також пасивні компоненти обв'язки (резистори серії C2-33 та конденсатори K10-17) для забезпечення фільтрації та електробезпеки користувача. Розробка принципової електричної схеми та проєктування топології друкованої плати буде здійснюватися у хмарному середовищі автоматизованого проєктування EasyEDA, яке дозволяє створити повний цифровий двійник пристрою з можливістю подальшого 3D-моделювання.

2.3.1 Розробка вузлу аналогового інтерфейсу (AFE)

Первинною ланкою в ланцюзі обробки біометричних даних є вузол аналогового інтерфейсу, реалізований на базі прецизійного інструментального підсилювача AD8232. Дана мікросхема спеціально спроектована для виділення, підсилення та фільтрації слабких біопотенціалів серця (ЕКГ-сигналів) на фоні значних шумів, спричинених рухом спортсмена та наводками електромережі.

Основне призначення вузла – перетворення диференціального сигналу, знятого з електродів, у чистий аналоговий сигнал з амплітудою, придатною для оцифрування АЦП мікроконтролера. Архітектура вузла включає в себе активний фільтр високих частот для усунення дрейфу ізолінії та фільтр низьких частот для відсікання м'язових шумів (електроміограми). Важливою особливістю схеми є наявність механізму захисту від перенапруг, що гарантує електробезпеку користувача та захищає вхідні каскади мікросхеми від статичного розряду.

В середовищі EasyEDA додаємо підсилювач AD8232. На вхід ставимо блок електродів – стандартний роз'єм з двома ніжками. Далі додаємо два резистори, які з'єднуємо з підсилювачем, та блок живлення 3.3В з конденсатором. Ці резистори утворюють бар'єр безпеки між тілом людини та джерелом живлення пристрою. Згідно з рекомендаціями виробника (Datasheet AD8232 [15]), для забезпечення електробезпеки користувача та захисту вхідних каскадів від статичних розрядів і дефібриляції, у ланцюг електродів включено обмежувальні резистори номіналом 180 кОм. Вибір даного номіналу обумовлений необхідністю обмеження струму через тіло пацієнта до рівня менше 20 мкА (згідно зі стандартом IEC 60601-1 [16]) у разі аварійної ситуації, при цьому зберігаючи високий вхідний імпеданс системи для якісної реєстрації ЕКГ-сигналу з амплітудою 0,5–2 мВ.

Також, відповідно до рекомендацій щодо проектування друкованих плат для прецизійних аналогових систем (Datasheet AD8232 [15]), паралельно

виводам живлення мікросхеми встановлено керамічний конденсатор ємністю 0,1 мкФ (100 нФ) з низьким значенням ESR. Даний компонент виконує роль блокувального конденсатора, що нівелює високочастотні шуми від роботи цифрових блоків мікроконтролера та радіомодуля Bluetooth 5.0, запобігаючи спотворенню корисного аналогового сигналу через лінії живлення. Розробка вузла у середовищі EasyEDA Pro показана на рисунку 2.4.

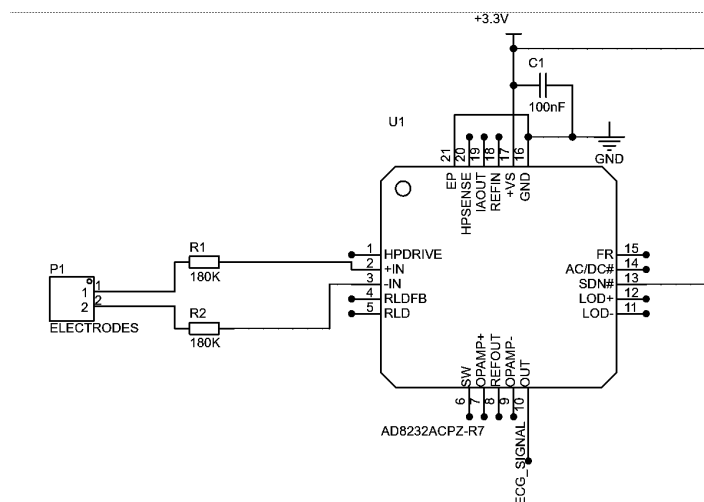


Рисунок 2.4 – Розробка вузла аналогового підсилювача

2.3.2 Мікроконтролер nRF52840

Для обробки та безпроводної передачі біометричних даних обрано систему на чіпі nRF52840 від компанії Nordic Semiconductor. Даний мікроконтролер базується на потужному 32-бітному ядрі ARM® Cortex®-M4 з тактовою частотою 64 МГц та має вбудований аналого-цифровий перетворювач (АЦП), що дозволяє з високою точністю оцифровувати сигнал з підсилювача AD8232.

Вибір nRF52840 обумовлений його рекордно низьким енергоспоживанням (до 5 мА в режимі передачі) та апаратною підтримкою

2.3.3 Антена та блок живлення

Для випромінювання сигналу Bluetooth 5.0 обрано малогабаритну керамічну патч-антену Taoglas SWLP.2450. Даний вибір обумовлений необхідністю мінімізації габаритів нагрудного датчика при збереженні високого коефіцієнту підсилення. Антена має багатоточкове заземлення (виводи 2-9), що дозволяє стабілізувати діаграму спрямованості та зменшити вплив тіла спортсмена на якість зв'язку. Узгодження виходу трансивера nRF52840 з антеною виконано за допомогою реактивних компонентів, що мінімізує коефіцієнт стоячої хвилі (VSWR) та забезпечує максимальну дальність передачі в режимі Long Range.

Для забезпечення автономної роботи нагрудного датчика обрано літєвий елемент живлення типорозміру CR2032 з номінальною напругою 3.0 В. Даний вибір обумовлений оптимальним співвідношенням енергоємності (близько 220 мАг) та габаритних розмірів, що дозволяє мінімізувати вагу пристрою.

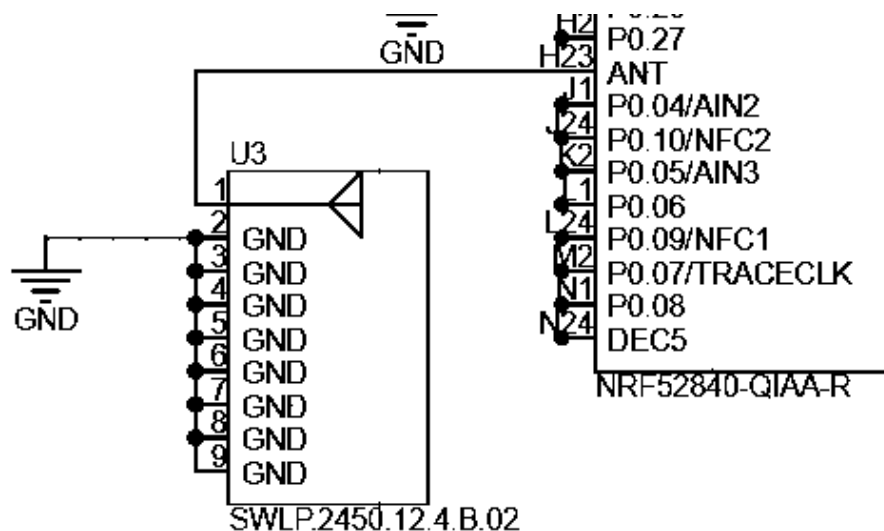


Рисунок 2.6 – Антена, під'єднана до мікроконтролеру

Вхідний каскад системи реалізований на базі інструментального підсилювача AD8232 (U1). Сигнал біопотенціалів серця надходить з електродної групи (P1) через обмежувальні резистори R1 та R2 номіналом 180 кОм.

Мікросхема U1 виконує підсилення сигналу та його очищення від артефактів руху за допомогою внутрішніх активних фільтрів. Для стабілізації роботи підсилювача використано блокувальний конденсатор C1 (100 нФ), а вхід Shutdown (пін 13) підтягнутий до шини живлення для постійної активації пристрою. Опрацьований аналоговий сигнал знімається з виходу OUT (пін 10) і через мітку ECG_SIGNAL передається на обчислювальний вузол.

Центральною частиною схеми є система на чіпі nRF52840 (U2). Мікроконтролер виконує оцифрування сигналу через 12-бітний АЦП (вхід P0.02). Живлення чіпа реалізовано за ієрархічною структурою: виводи VDD та VDDH об'єднані в єдину шину 3.3 В. Для стабілізації напруги внутрішнього ядра використано конденсатор C3 (1 мкФ) на виводі DEC4.

Тактування радіочастотного модуля забезпечується пасивним кварцовим резонатором X1 (32 МГц), який працює в парі з навантажувальними конденсаторами для забезпечення прецизійної стабільності частоти 2.4 ГГц.

Для безпроводної передачі даних за протоколом Bluetooth 5.0 Long Range використано вихід ANT (пін H23), підключений до керамічної патч-антени SWLP.2450 (U3). Багатоточкове заземлення антени (піни 2-9) забезпечує стабільність діаграми спрямованості та мінімізує вплив тіла спортсмена на якість зв'язку.

Живлення пристрою здійснюється від літієвого елемента CR2032 (B1). Схема передбачає механічний вимикач SW1, що забезпечує повне знеструмлення приладу в режимі зберігання. Фільтрація загальної шини живлення виконується конденсатором C2 (100 нФ), що нівелює високочастотні наводки в колах живлення.

На завершальному етапі розробки принципової схеми було проведено верифікацію за допомогою інструменту Design Rule Check (DRC). Результати перевірки (0 Fatal Errors, 0 Errors) підтвердили коректність електричних з'єднань.

Слід зауважити, що представлена принципова електрична схема є функціонально-еквівалентною. З метою підвищення читабельності та акцентування уваги на архітектурних рішеннях системи, внутрішні структури інтегрованих модулів (таких як складні каскади підсилення AD8232, що містять десятки операційних підсилювачів та транзисторів, або високочастотний радіотракт nRF52840) представлені у вигляді уніфікованих графічних блоків. Такий підхід відповідає стандартам проєктування складних мікроелектронних систем, де розробник оперує готовими системними рішеннями, фокусуючись на міжкомпонентних зв'язках, вузлах узгодження та цілісності сигналів, залишаючи внутрішню напівпровідникову топологію чіпів на рівні виробника інтегральних схем.

2.3.5 Програмне забезпечення для датчику пульсу

Оскільки ми використовуємо nRF52840, код пишеться на мові C++ з використанням бібліотек для Bluetooth Low Energy (BLE).

Програмне забезпечення нагрудного датчика базується на подієво-орієнтованій архітектурі та використовує стек протоколів Bluetooth Low Energy (BLE) 5.0. Основна логіка роботи реалізована у вигляді циклічного опитування вбудованого 12-бітного аналого-цифрового перетворювача (АЦП), який зчитує напругу з виходу підсилювача AD8232.

Програма виконує наступні ключові функції:

1. Цифрова обробка сигналу: Зчитані значення АЦП проходять через програмний алгоритм детекції піків (R-зубців ЕКГ), що дозволяє розрахувати частоту серцевих скорочень (BPM) у реальному часі.
2. BLE Advertising: Пристрій транслює унікальний ідентифікатор (Athlete_ID), що дозволяє системі розрізняти дані від різних спортсменів.
3. Трансляція даних: Для забезпечення максимальної сумісності та низького енергоспоживання, передача даних здійснюється через стандартний сервіс Heart Rate Service (UUID 0x180D).
4. Оптимізація живлення: Після кожної ітерації вимірювання мікроконтролер переходить у режим низького енергоспоживання (System ON sleep), що дозволяє мінімізувати струм розряду батареї.

З метою уніфікації апаратного забезпечення та спрощення логістики під час масових змагань, у системі реалізовано принцип динамічної ідентифікації. Кожен нагрудний датчик має унікальний апаратний ідентифікатор (Device ID), який формується на основі заводського 64-бітного серійного номера мікроконтролера nRF52840 (реєстри DEVICEADDR).

Процес співставлення даних із конкретним учасником змагань перенесено на рівень прикладного програмного забезпечення WEB-системи. У базі даних буде реалізовано таблицю зв'язків, яка дозволяє оператору в реальному часі закріплювати ідентифікатор пристрою за конкретним спортсменом на час проведення забігу. Це дозволяє використовувати обмежений парк датчиків для необмеженої кількості учасників, забезпечуючи при цьому повну цілісність та персоналізацію зібраних телеметричних даних.

Повний код наведений у додатку Б.

2.4 Розробка приймального шлюзу

Приймальний шлюз є центральною сполучною ланкою системи, що реалізує концепцію граничних обчислень. Основна логіка роботи пристрою полягає у перетворенні протоколів передачі даних: від енергоефективного персонального радіоканалу Bluetooth 5.0 Long Range (Coded PHY) до універсального мережевого протоколу Wi-Fi (IEEE 802.11 b/g/n).

Для розробки шлюзу ми використаємо мікроконтролер з вбудованим кварцом та антенами ESP32-S3-WROOM-1, блок індикації, блок живлення.

2.4.1 Мікроконтролер ESP32

Для реалізації функцій приймального шлюзу обрано систему на чіпі ESP32-S3 у виконанні готового сертифікованого модуля ESP32-S3-WROOM-1. Даний мікроконтролер базується на потужному двоядерному 32-бітному процесорі Xtensa® LX7 з тактовою частотою до 240 МГц. Ключовою перевагою обраної модифікації є апаратна підтримка стандарту Bluetooth 5.0 з функцією Long Range (Coded PHY), що забезпечує повну сумісність із розробленим нагрудним датчиком та дозволяє приймати телеметричні дані на значних відстанях.

У структурі системи ESP32-S3 виконує роль інтелектуального шлюзу. Завдяки наявності вбудованого Wi-Fi модуля стандарту 802.11 b/g/n, пристрій забезпечує стабільну ретрансляцію отриманих даних у WEB-систему за протоколом HTTP. Використання модуля (WROOM) замість окремого чіпа дозволяє значно спростити принципову схему шлюзу, оскільки радіочастотний тракт, антена та кварцовий резонатор уже інтегровані

всередині модуля та захищені металевим екраном від електромагнітних завад. Вигляд мікроконтролеру у середовищі EasyEDA Pro показана на рисунку 2.9.

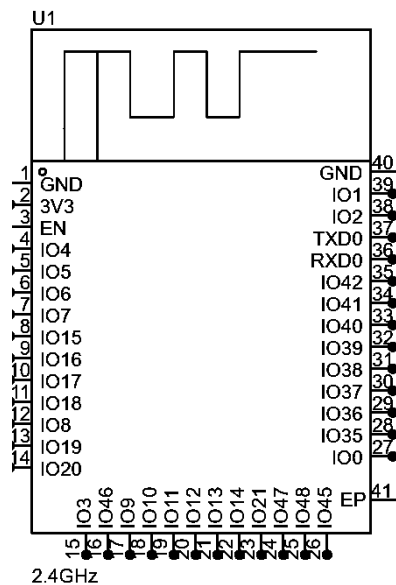


Рисунок 2.9 – Мікроконтролер ESP32-S3-WROOM-1-N8

2.4.2 Блок живлення для мікроконтролеру приймального шлюзу

Для забезпечення стабільної роботи шлюзу в умовах пікових навантажень (активація Wi-Fi модему), використано схему стабілізації на базі лінійного регулятора з низьким падінням напруги AMS1117-3.3. Вхідний каскад розрахований на напругу +5 В, що дозволяє використовувати стандартні USB-джерела живлення. Для придушення високочастотних завад та компенсації імпульсних просідань напруги, у ланцюг живлення включено танталові конденсатори ємністю 10 мкФ. Мікроконтролер ESP32-S3 переведено в активний режим шляхом підтяжки виводу EN до шини живлення 3.3 В. Блок живлення у середовищі EasyEDA Pro показано на рисунку 2.10.

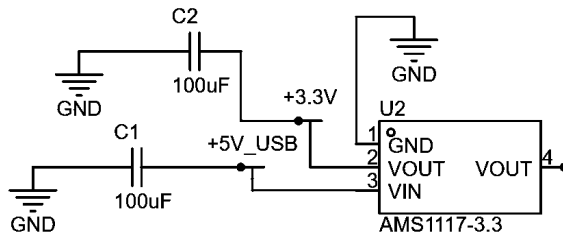


Рисунок 2.10 – Блок живлення для мікроконтролеру

2.4.3 Блок індикації

Для забезпечення оперативного контролю за станом роботи приймального шлюзу в реальному часі до складу схеми включено вузол візуальної індикації. Він побудований на базі малогабаритного світлодіода червоного кольору світіння (LED1), підключеного до універсального порту введення-виведення GPIO 1 мікроконтролера ESP32-S3. Основною функцією даного вузла є сигналізація про успішне отримання пакетів даних від нагрудних датчиків та підтвердження активності Wi-Fi з'єднання.

Технічна реалізація вузла передбачає використання обмежувального резистора R1 номіналом 220 Ом, що включений послідовно в електричне коло. Роль даного компонента полягає у стабілізації робочого струму світлодіода на рівні 6–8 мА, що оберігає напівпровідникову структуру індикатора від перегріву, а вихідний каскад мікроконтролера – від перевантаження за струмом. Програмна логіка керування індикацією побудована на принципі короточасних імпульсів (миготіння), що дозволяє візуально диференціювати цикли прийому-передачі даних та знизити загальне енергоспоживання пристрою.

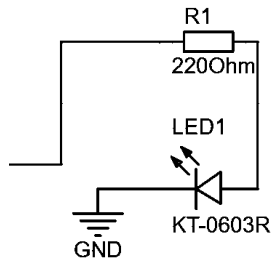


Рисунок 2.11 – Блок індикації

Для обмеження струму через індикаторний світлодіод розраховано номінал обмежувального резистора R_{lim} . При вихідній напрузі логічної одиниці мікроконтролера $U_{oh} = 3.3V$ та прямому падінні напруги на червоному світлодіоді $U_f \approx 1.8V$, необхідний опір для забезпечення робочого струму $I_f = 7$ мА складає:

$$R = \frac{U_{oh} - U_f}{I_f} = \frac{3.3 - 1.8}{0.007} \approx 214 \text{ Ом}$$

Обрано стандартне значення з ряду E24 – 220 Ом.

2.4.4 Принципова електрична схема приймального шлюзу

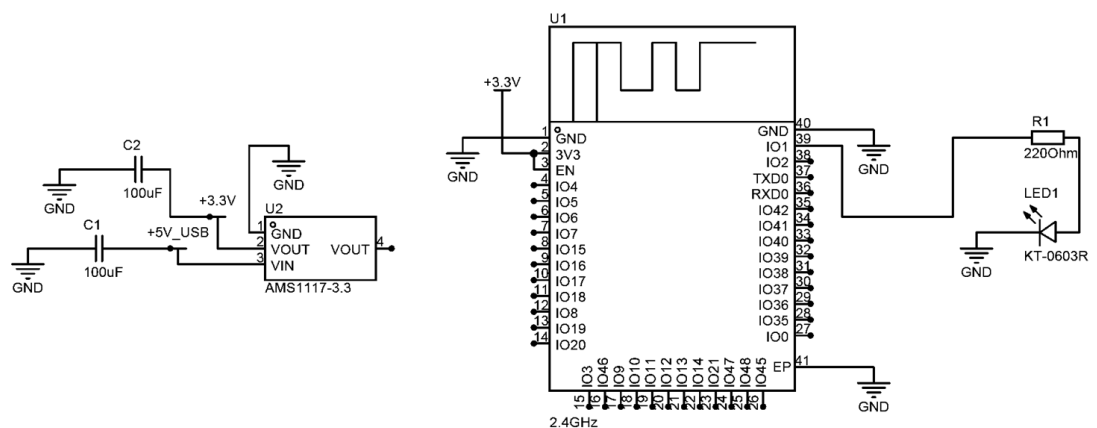


Рисунок 2.12 – Принципова електрична схема приймального шлюзу

Розроблена принципова електрична схема приймального шлюзу (рис. 2.11) призначена для забезпечення безперебійного зв'язку між персональною мережею датчиків та глобальною мережею Internet. Пристрій побудований за модульним принципом і включає такі ключові вузли:

1. Центральний обчислювальний модуль

Основою вузла є мікроконтролерний модуль ESP32-S3-WROOM-1 (U1). Він інтегрує в собі двоядерний процесор, Wi-Fi антену та радіомодуль Bluetooth 5.0. Пін 3V3 (2) підключений до шини живлення, а вивід EN (3) підтягнутий до високого логічного рівня для автоматичного старту програми при подачі напруги. Заземлення реалізовано через піни GND (1, 40) та центральний термальний майданчик EP (41), що також забезпечує відвід тепла від чіпа.

2. Система стабілізації живлення

Вузол живлення реалізований на базі лінійного регулятора напруги AMS1117-3.3 (U2). Вхідна напруга +5 В (мітка +5V_USB) надходить на пін VIN (3), а стабілізована напруга +3.3 В знімається з піна VOUT (2). Для компенсації імпульсних просідань напруги, спричинених роботою Wi-Fi передавача, у схему включено два фільтруючі конденсатори C1 та C2 ємністю 100 мкФ. Такий номінал забезпечує високу стабільність живлення та захищає мікроконтролер від перезавантажень.

3. Вузол світлодіодної індикації

Для візуального моніторингу активності пристрою використано світлодіод LED1 (KT-0603R). Він підключений до універсального виводу IO1 (пін 39). Послідовно з індикатором встановлено обмежувальний резистор R1 номіном 220 Ом, який задає безпечний робочий струм у діапазоні 6–8 мА. Програмна логіка передбачає миготіння світлодіода при кожному успішному отриманні та ретрансляції даних.

2.4.5 Розробка програмного забезпечення мікроконтролера приймального шлюзу

Програмне забезпечення приймального шлюзу реалізовано на базі фреймворку Arduino ESP32. Логіка ПЗ розділена на два функціональні потоки.

Перший потік відповідає за роботу радіомодуля в режимі BLE Central. Він здійснює сканування ефіру, ідентифікацію датчиків за маскою імені UA-SENSOR– та встановлення з'єднання в режимі Long Range (Coded PHY). Після підключення шлюз підписується на сповіщення характеристики 0x2A37 (Heart Rate Measurement).

Другий потік забезпечує роботу Wi-Fi стеку та взаємодію з WEB-системою. При отриманні даних від датчика, програма формує JSON-об'єкт, що містить унікальний ідентифікатор пристрою та значення пульсу. Передача здійснюється методом HTTP POST на віддалений сервер. Для візуального контролю процесу реалізовано алгоритм індикації: короткочасна активація світлодіода (GPIO 1) сигналізує про успішний цикл «прийом-передача».

Повний код наведено у додатку В.

Для верифікації розробленого програмного забезпечення було створено віртуальний стенд у середовищі моделювання Wokwi. Оскільки фізичне з'єднання за протоколом Bluetooth 5.0 Long Range у браузерному середовищі обмежене, було застосовано метод імітаційного моделювання вхідних даних. Спеціально розроблений тестовий сценарій генерував випадкові значення пульсу в діапазоні 70–160 уд/хв, які оброблялися мережевим стеком ESP32-S3. Успішна ретрансляція JSON-об'єктів на тестовий сервер (httpbin.org) та коректна робота вузла індикації підтвердили працездатність алгоритму та готовність системи до розгортання в реальних умовах. Скріншот з симуляції наведено на рисунку 2.13.

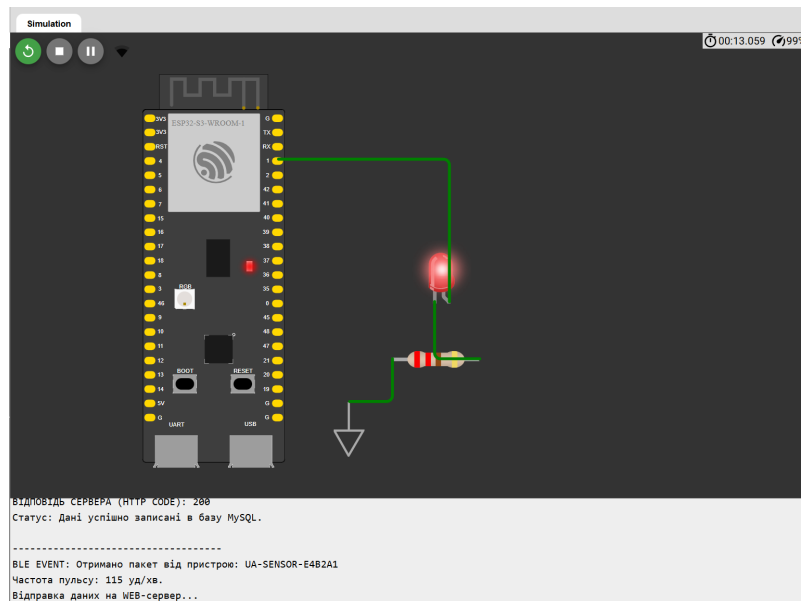


Рисунок 2.13 – Результати дослідження параметрів приймального шлюзу у середовищі Wokwi

2.5 Розробка WEB-системи

Завершальним етапом створення комплексної мікроелектронної системи є розробка спеціалізованого веб-орієнтованого програмного забезпечення, що виконує функції верхнього рівня архітектури IoT. Основною метою даної підсистеми є централізований збір, довгострокове зберігання та інтерактивна візуалізація телеметричних даних, отриманих від приймальних шлюзів у реальному часі. Веб-система виступає в ролі консолідуючого центру, який перетворює сирі потоки даних від датчиків на структуровану інформацію, доступну для аналізу тренерами, суддями та глядачами змагань. Враховуючи необхідність високої швидкості обробки запитів, масштабованості та надійності зберігання результатів, архітектура системи базується на сучасних веб-технологіях, що забезпечують безперебійну роботу з великими об'ємами даних та гнучке управління правами доступу.

2.5.1 Проектування структури та архітектури інформаційної системи

Існує безліч підходів до проектування інформаційних web-систем. Найбільш популярними та загальновживаними шаблонами є MVC (Model-View-Controller), MVVM (Model-View-ViewModel), MVP (Model-View-Presenter) та інші. В рамках розробки дипломного проєкту я зупинився на використанні MVC як такої схеми проектування, що надає можливість побудувати масштабований веб-додаток з чітко розділеною логікою виконання.

Model-View-Controller (MVC, «Модель-Представлення-Контролер») – схема поділу даних програми, призначеного для користувача інтерфейсу і керуючої логіки на три окремих компоненти: модель, уявлення і контролер – таким чином, що модифікація кожного компонента може здійснюватися незалежно [17]. Можна виділити такі загальні ролі кожного компоненту:

- модель (Model) надає дані і реагує на команди контролера, змінюючи свій стан;
- представлення (View) відповідає за відображення даних моделі користувачеві, реагуючи на зміни моделі;
- контролер (Controller) інтерпретує дії користувача, сповіщаючи модель про необхідність змін.

Основна мета застосування концепції MVC полягає в відділенні бізнес-логіки (моделі) від її візуалізації (представлення, виду). За рахунок такого поділу підвищується можливість повторного використання коду. Найбільш корисне застосування даної концепції в тих випадках, коли користувач повинен бачити ті ж самі дані одночасно в різних контекстах і / або з різних точок зору. До однієї моделі можна приєднати кілька виглядів, при цьому не зачіпаючи реалізацію моделі. Наприклад, деякі дані можуть бути одночасно представлені у вигляді електронної таблиці, гістограми і кругової діаграми.

Враховуючи суть моделі MVC, можна скласти узагальнену структурну схему розроблюваної системи. Ця схема подана на рисунку 2.14.

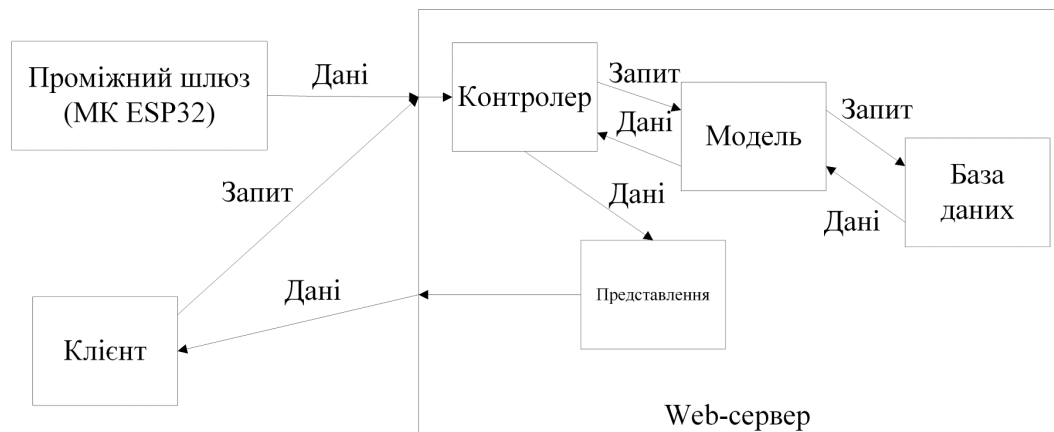


Рисунок 2.14 – Структурна схема інформаційної web-системи

2.5.2 Алгоритм роботи web-системи

Перш за все, користувач може заходити на сайт або як його працівник, або як звичайний відвідувач. Тобто він може запросити або адміністративну частину, або будь-який інший ресурс системи. Тому спочатку потрібно перевірити, у який розділ сервісу він намагається отримати доступ.

Якщо користувач хоче потрапити до адміністративного розділу, він повинен пройти авторизацію. В разі успішності він отримує доступ до інтерфейсу адміністратора.

Якщо відвідувач просто хоче отримати певну інформацію на сайті, його запит (в конкретному випадку – URL запити) аналізує контролер сайту. Якщо ресурс, на який робиться запит, не існує, то користувача переадресує на сторінку, що повертає http-код 404. У випадку успіху запит передається спеціалізованому контролеру того чи іншого розділу сайту, який у свою чергу звертається до моделі за даними. Модель повертає дані з БД, які потім передаються представленню для візуалізації інформації для користувача.

Загальний алгоритм роботи системи поданий на рисунку 2.15.

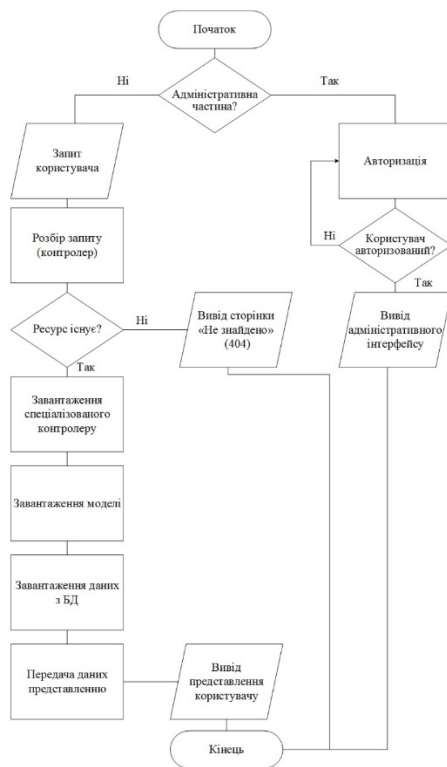


Рисунок 2.15 – Алгоритм роботи інформаційної web-системи

Пристрій введення інформації можна представити у вигляді електричної схеми на рис. 2.16.

Шифратор клавіатури виконано на трьох мікросхемах K155ЛЛ1 з відповідною комутацією. Отриманий в результаті перетворень двійковий код клавіші, яку натискають, надходить на формувач сигналу готовності «Готовий», який складається з елемента 5АБО (у спрощеному вигляді – 4 елемента 2АБО), який будується на мікросхемі K155ЛЛ1, інвертора (з мікросхеми K155ЛН1) та елемента «виключаюче АБО» (на мікросхемі K155ЛП5). Буферний регістр на виходах клавіатури побудований на тригерах, які виконані на двох мікросхемах K155ТМ2. Тригери управляються імпульсами, які виробляє генератор, зібраний на мікросхемі K155ЛА3. Далі код поступає на схему контролю парності на мікросхемах K155ЛП5 і по лінії зв'язку передається на наступний пристрій – перетворювач двійкового коду в код «Із надлишком 4».

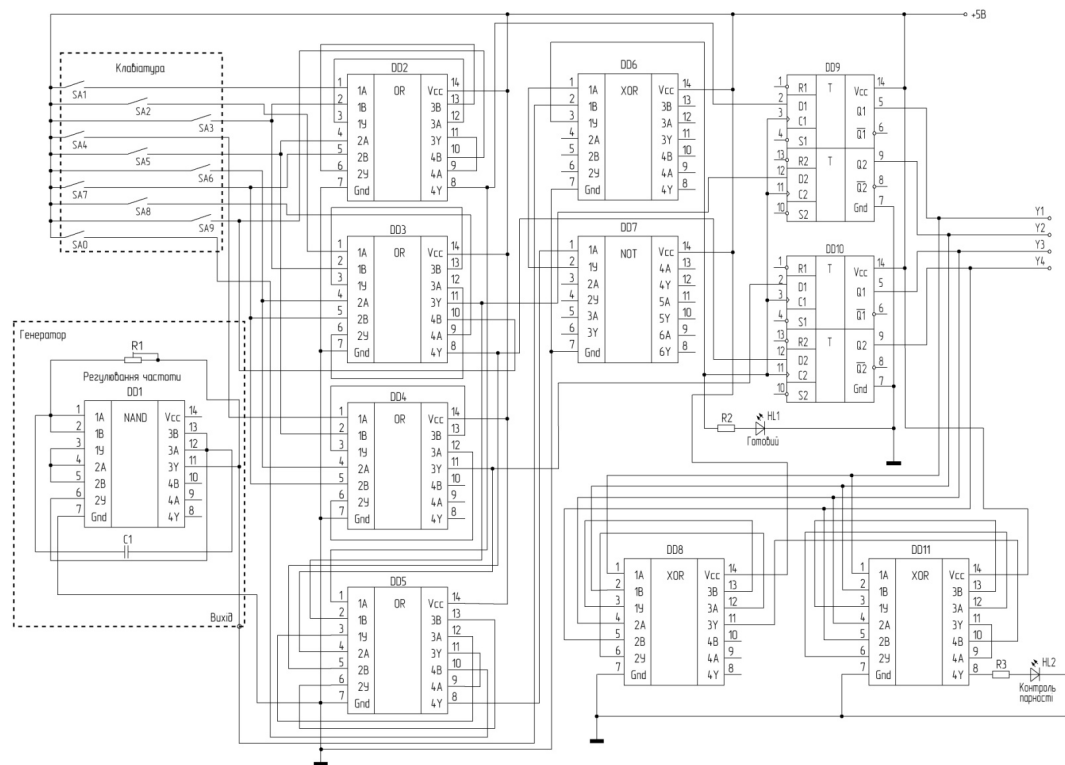


Рисунок 2.16 – Схема електрична принципова введення інформації

Після того, як розробник має уявлення про загальну структуру проєкту, можна переходити до проєктування БД сайту.

2.5.3 Проєктування бази даних інформаційної системи

Постійний запам'ятовуючий пристрій (ПЗП) – незалежні пристрої, тобто зберігають інформацію навіть при вимкненні електроживлення. ПЗП, як правило, реалізуються на основі діодної матриці, в якій діоди розташовуються на пересічення горизонтальних та вертикальних шин, що є лініями стовпців [8; 11]. Інформація прочитується з розрядних шин. Логічна одиниця з'явиться на тих розрядних шинах, які через діод підключені до адресної шини. Функціональна схема ПЗП зображена на рис. 2.17.

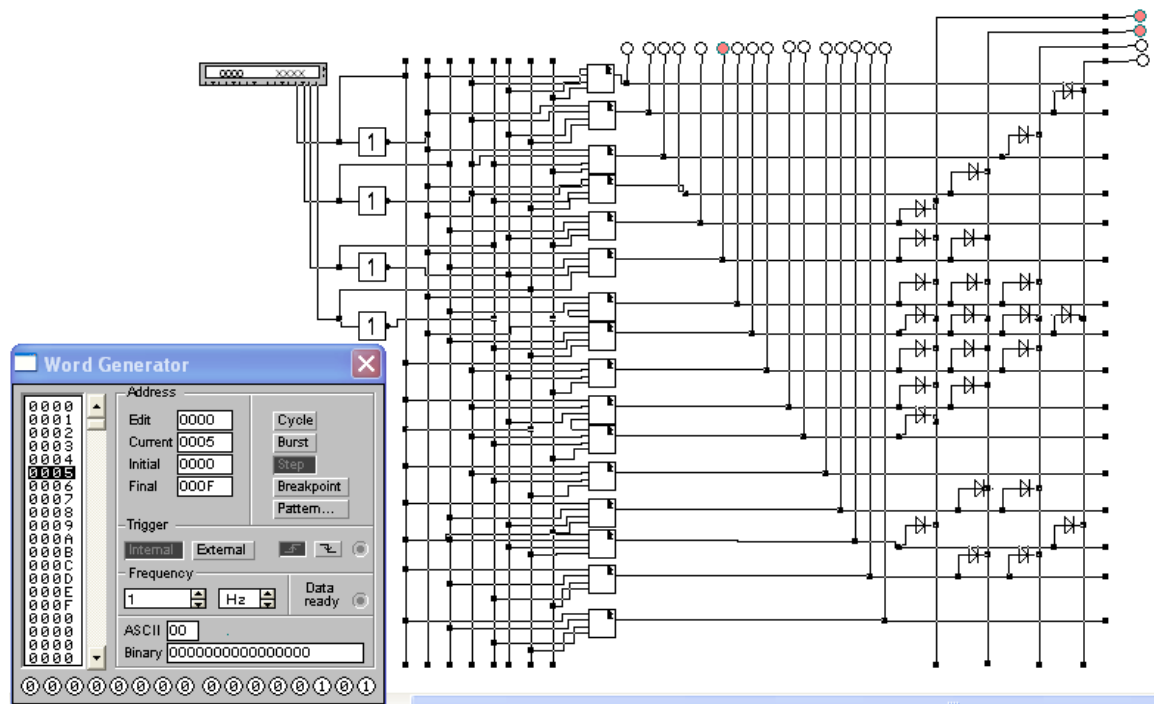


Рисунок 2.17 – Функціональна схема ПЗП

Перший етап проєктування бази даних називається концептуальним моделюванням бази даних. Концептуальне проєктування – створення представлення бази даних, що включає визначення типів найважливіших сутностей і існуючих між ними зв'язків і атрибутів. Концептуальне моделювання бази даних полягає в конструюванні інформаційної моделі системи, що не залежить від будь-яких фізичних умов реалізації.

На підставі аналізу предметної області можна виділити сутності, інформація про які зберігається у базі даних:

- сутність Пульсометри;
- сутність Спортсмени;
- сутність Види спорту;
- сутність Змагання;
- сутність Користувач;
- сутність Групи Користувачів.

Існування цих сутностей не залежить від існування інших сутностей, тому ці сутності є базовими.

Асоціативні сутності – це Призначення пристроїв (пульсометр спортсмена на конкретному змаганні), а також Телеметрія (пульс спортсмена).

Підсумкова ER-діаграма бази даних надана на рисунку 2.18.

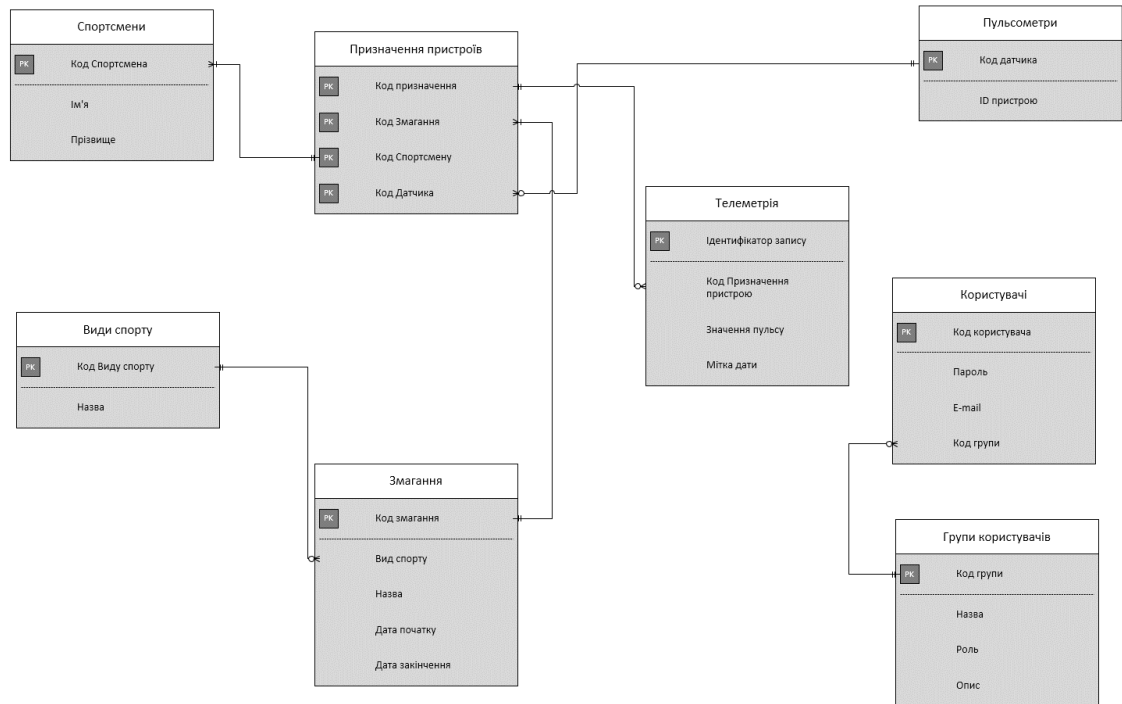


Рисунок 2.18 – ER-діаграма бази даних інформаційної web-системи

Наступний етап – логічне моделювання бази даних. Логічне проєктування бази даних являє собою процес конструювання моделі інформаційної структури організації, що виконується на основі конкретної моделі даних, але без урахування певної СУБД і інших фізичних обмежень.

Згідно діаграми визначаємо, що у підсумковій базі даних має бути 8 відношень: Пульсометри, Спортсмени, Види спорту, Змагання, Призначення пристроїв, Телеметрія, Користувачі, Групи користувачів. Кожна таблиця була проаналізована на предмет:

- Визначення списку атрибутів для кожного відношення;
- Для кожного атрибуту визначені умови цілісності даних та обов'язковість наявності даних (припустимість значення NULL);

- Обмеження для доменів атрибутів;
- Тип даних для кожного атрибуту відношення.

Результати аналізу наведені у таблицях, складеній для кожного відношення.

Таблиця 2.1 – Відношення «Спортсмени»

Атрибут	Тип даних	Припустиме значення	Обов'язковість	Примітка
Код Спортсмена	Лічильник	Натуральні числа	Так	Первинний ключ
Ім'я	Рядок	Кирилиця, Латиниця	Так	-
Прізвище	Рядок	Кирилиця, Латиниця	Так	-

Таблиця 2.2 – Відношення «Види спорту»

Атрибут	Тип даних	Припустиме значення	Обов'язковість	Примітка
Код виду спорту	Лічильник	Натуральні числа	Так	Первинний ключ
Назва	Рядок	Кирилиця, Латиниця	Так	-

Таблиця 2.3 – Відношення «Пульсометри»

Атрибут	Тип даних	Припустиме значення	Обов'язковість	Примітка
Код датчику	Лічильник	Натуральні числа	Так	Первинний ключ
ID пристрою	Рядок	Латиниця	Так	-

Таблиця 2.4 – Відношення «Змагання»

Атрибут	Тип даних	Припустиме значення	Обов'язковість	Примітка
Код Змагання	Лічильник	Натуральні числа	Так	Первинний ключ
Вид спорту	Число	Натуральні числа	Так	-
Назва	Рядок	Кирилиця, Латиниця	Так	-
Дата початку	Дата/час	Дата/час	Так	-
Дата завершення	Дата/час	Дата/час	Так	-

Таблиця 2.5 – Відношення «Призначення пристроїв»

Атрибут	Тип даних	Припустиме значення	Обов'язковість	Примітка
Код Призначення	Лічильник	Натуральні числа	Так	Первинний ключ
Код змагання	Число	Натуральні числа	Так	Первинний ключ
Код спортсмену	Число	Натуральні числа	Так	Первинний ключ
Код датчику	Число	Натуральні числа	Так	Первинний ключ

Таблиця 2.6 – Відношення «Телеметрія»

Атрибут	Тип даних	Припустиме значення	Обов'язковість	Примітка
Ідентифікатор запису	Лічильник	Натуральні числа	Так	Первинний ключ
Код призначення	Число	Натуральні числа	Так	-
Значення пульсу	Число	Натуральні числа	Так	-
Мітка дати	Дата/час	Дата/час	Так	-

Таблиця 2.7 – Відношення «Користувачі»

Атрибут	Тип даних	Припустиме значення	Обов’язковість	Примітка
Код Користувача	Лічильник	Натуральні числа	Так	Первинний ключ
Пароль	Рядок	Латиниця	Так	—
E-mail	Рядок	Латиниця	Так	—
Код Групи	Число	Натуральні числа	Так	—

Таблиця 2.8 – Відношення «Групи користувачів»

Атрибут	Тип даних	Припустиме значення	Обов’язковість	Примітка
Код Групи	Лічильник	Натуральні числа	Так	Первинний ключ
Назва	Рядок	Латиниця	Так	—
Роль	Рядок	Кирилиця	Так	—
Опис	Рядок	Кирилиця	Так	—

Проектування структури бази даних виконано з дотриманням принципів реляційної алгебри, що забезпечує відповідність усіх відношень третій нормальній формі (3NF) та нормальній формі Бойса-Кодда (BCNF). Кожна таблиця має унікальний первинний ключ, а всі неключові атрибути мають повну функціональну залежність виключно від основного ключа, що виключає наявність транзитивних залежностей.

2.5.4 Розробка програмного забезпечення інформаційної web-системи

Першочерговим етапом проектування web-інтерфейсу є чітке визначення необхідних розділів та сторінок. Згідно з технічним завданням, інформаційна система містить у собі 2 основні частини – гостьову та адміністративну. На гостьовій частині на даний момент користувачі зможуть

слідкувати за показниками пульсу спортсменів обраних змагань у режимі реального часу. У свою чергу, в адміністративній частині буде відбуватись керування змаганнями, прив'язкою спортсменів до датчиків і так далі. Схема web-інтерфейсу інформаційної системи наведена на рисунку 2.19.



Рисунок 2.19 – Схема веб-інтерфейсу інформаційної системи

Для реалізації веб-системи обрано фреймворк Laravel 10 (PHP 8.1), що забезпечує надійну архітектуру за шаблоном MVC (Model-View-Controller). Для фронтенд-частини застосовано прогресивний JS-фреймворк Vue.js через інструмент Inertia.js, що дозволяє створювати Single Page Application (SPA) без необхідності розробки складного API.

Для початку у фреймворку Laravel за допомогою інструменту міграцій було створено необхідні бази даних. Після цього база даних має наступний вигляд (скріншот з середовища phpMyAdmin – програми для зручної роботи з базами даних – рисунок 2.20):

Таблиця	Дія	Рядки	Тип	Зіставлення	Розмір	Фрагментовані
☐ assignments	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	64.0 KB	-
☐ athletes	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
☐ competitions	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
☐ failed_jobs	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
☐ migrations	★ Переглянути Структура Пошук Вставити Очистити Знищити	11	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
☐ password_reset_tokens	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
☐ personal_access_tokens	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
☐ sensors	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
☐ sports	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
☐ telemetry	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
☐ users	★ Переглянути Структура Пошук Вставити Очистити Знищити	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
11 таблиць	Всього	11	InnoDB	utf8mb4_general_ci	336.0 KB	0 B

Рисунок 2.20 – Таблиці бази даних

Для забезпечення взаємодії між мікроконтролерним пристроєм (ESP32) та веб-системою було розроблено REST API інтерфейс. Прийом даних здійснюється через HTTP POST запити у форматі JSON. Логіка обробки вхідних даних передбачає ідентифікацію датчика за його унікальним апаратним номером (`device_id`), пошук активного призначення (зв'язок датчика зі спортсменом на конкретному змаганні) та фіксацію показника частоти серцевих скорочень (`bpm`) у таблиці телеметрії з автоматичною міткою часу. Код контролеру обробки сигналу від мікроконтролеру приймального шлюзу наведено у додатку Г.

Адміністративна частина системи реалізована за допомогою контролерів, що забезпечують CRUD-функціонал (Create, Read, Update, Delete). Використання Inertia.js дозволяє передавати дані з серверної частини (Laravel) до клієнтської (Vue.js) як пропси (props), зберігаючи при цьому звичну маршрутизацію фреймворку.

Для візуалізації даних телеметрії в реальному часі на фронтенд-частині використано компонентну архітектуру Vue.js 3. Для побудови графіків частоти серцевих скорочень обрано бібліотеку Chart.js, яка дозволяє динамічно відображати зміни пульсу спортсмена залежно від часу змагання.

У системі реалізовано рольову модель управління доступом. Для запобігання несанкціонованому доступу, можливість публічної реєстрації було відключено. Початковий доступ до системи отримує адміністратор, обліковий запис якого створюється програмним шляхом на рівні сервера. В подальшому адміністратор має виключне право на створення нових

користувачів та призначення їм відповідних ролей (тренер, спортсмен тощо). Для забезпечення гнучкого управління правами доступу було реалізовано окрему сутність «Ролі користувачів». Це дозволяє масштабувати систему: адміністратор може створювати нові групи (наприклад, «Тренер», «Спортсмен») без зміни коду програми. Кожен користувач системи жорстко прив'язаний до ідентифікатора ролі, що визначає його рівень доступу до функціоналу. Інтерфейс розділу надано на рисунку 2.21.

Dashboard Головний Адмін ▾

Управління групами (ролями)

Створити нову групу

Системна назва (name)

Відображувана назва (title)

Додати групу

Список зареєстрованих груп

ID	Системна назва	Назва групи
1	admin	Адміністратор
2	athlete	Спортсмен
3	coach	Тренер

Рисунок 2.21 – Створення ролей користувачів адміністратором

Модуль управління користувачами дозволяє адміністратору створювати нові облікові записи, прив'язуючи їх до відповідних ролей. Паролі користувачів хешуються за допомогою алгоритму BCrypt перед збереженням у базі даних, що відповідає сучасним стандартам безпеки інформаційних систем. Взаємодія з базою даних здійснюється через ORM Eloquent, що мінімізує ризики SQL-ін'єкцій. Для підвищення надійності системи реалізовано дворівневу перевірку вхідних даних. На рівні сервера (Laravel Validation) виконується перевірка на відповідність форматів (валідація email, мінімальна довжина пароля) та унікальність записів у базі даних. У разі виникнення помилок, сервер повертає JSON-об'єкт з описами порушень, які динамічно відображаються користувачеві в інтерфейсі за допомогою

реактивних властивостей Vue.js. Приклад коду контролеру та Vue-компоненту представлено у додатках Д та Е. Скріншот розділу користувачів надано на рисунку 2.22.

Управління користувачами

Зареєструвати нового користувача

ІМ'Я

EMAIL

ПАРОЛЬ

ГРУПА ДОСТУПУ

Виберіть роль

Створити

Список користувачів у системі

Ім'я	Email	Група	Дата реєстрації	Дії
Головний Адмін	admin@test.com	Адміністратор	02.01.2026	
Другий адмін	admin@admin2.ua	Адміністратор	02.01.2026	Видалити

Рисунок 2.22 – Управління користувачами

Для розмежування прав доступу використано механізм Middleware (проміжне програмне забезпечення). Створено фільтр доступу, який перевіряє роль користувача перед наданням доступу до адміністративних маршрутів. Це гарантує, що лише користувачі з групою «admin» мають право керувати обліковими записами та налаштуваннями системи.

Управління апаратним забезпеченням системи здійснюється через модуль «Датчики». Кожен датчик ідентифікується унікальним рядковим кодом (Device ID), який програмується безпосередньо в мікроконтролер датчику пульсу. Система забезпечує можливість реєстрації нових пристроїв та їх видалення з бази даних у разі виведення з експлуатації. Інтерфейс наведено на рисунку 2.23.

Додати датчик

Device ID

Модель

Додати

ID	Device ID	Модель	Дії
1	UA-SENSOR-1515515	test	Редагувати Видалити

Рисунок 2.23 – Сторінка роботи з датчиками пульсу

Модуль «Спортсмени» призначений для реєстрації учасників змагань. Система зберігає персональні дані (прізвище та ім'я), які в подальшому пов'язуються з датчиками телеметрії. Інтерфес показано на рисунку 2.24.

Спортсмени

Додати спортсмена

ІМ'Я

ПРІЗВИЩЕ

Зберегти

ID	Прізвище	Ім'я	Дії
1	Петров	Петро	Редагувати Видалити
2	Степанов	Степан	Редагувати Видалити

Рисунок 2.24 – Модуль для роботи зі списком спортсменів

Розділ «Види спорту» є базовою класифікаційною сутністю системи. Він дозволяє групувати змагання за напрямками (наприклад: легка атлетика, велоспорт, плавання). Така структура забезпечує можливість подальшого розширення системи та специфікації алгоритмів обробки телеметрії залежно від обраної дисципліни. Інтерфейс надано на рисунку 2.25.

Довідник: Види спорту

Додати новий вид спорту

Напр: Легка атлетика

Додати

ID	НАЗВА ДИСЦИПЛІНИ	Дії
1	Легка атлетика	Ред. Видалити
2	Футбол	Ред. Видалити

Рисунок 2.25 – Розділ «Види спорту»

Модуль «Змагання» є центральним організаційним вузлом системи. Він агрегує дані про вид спорту, назву заходу та його часові межі (час початку та завершення). Реалізація цього модуля передбачає використання зв'язків «багато до одного» з таблицею видів спорту. Цей розділ дозволяє адміністратору планувати спортивні заходи та готувати систему до прийому телеметричних даних. Інтерфейс модулю наданий на рисунку 2.26.

Управління змаганнями

Створити нове змагання

ВИД СПОРТУ:
 НАЗВА:
 ПОЧАТОК:
 ЗАВЕРШЕННЯ:

Назва	Вид спорту	Початок	Кінець	Дії
Всеукраїнські змагання	Легка атлетика	2026-01-03 10:00:00	2026-01-03 21:00:00	Ред. Видалити
Міські змагання	Легка атлетика	2026-01-10 09:00:00	2026-01-10 19:05:00	Ред. Видалити

Рисунок 2.26 – Розділ роботи зі змаганнями

Модуль «Призначення пристроїв» виступає сполучною ланкою між апаратною та програмною частинами системи. Він реалізує логічний зв'язок між трьома ключовими сутностями: конкретним змаганням, спортсменом та фізичним пристроєм (датчиком). Саме завдяки цій структурі система ідентифікує вхідні JSON-пакети від мікроконтролерів: отримуючи `device_id`, сервер через таблицю призначень визначає, якому спортсмену належить цей показник пульсу в межах поточного заходу. Інтерфес розділу – на рисунку 2.27.

Призначення пристроїв

Нова прив'язка "Спортсмен - Датчик"

ЗМАГАННЯ

СПОРТСМЕН

ДАТЧИК (DEVICE ID)

Оберіть змагання...

Оберіть спортсмена...

Оберіть пристрій...

Призначити

Змагання	Спортсмен	Датчик	Дія
Всеукраїнські змагання	Петров Петро	UA-SENSOR-1515515	Видалити

Рисунок 2.27 – Розділ роботи зі змаганнями

Для верифікації працездатності системи на етапі розробки було створено модуль імітаційного моделювання телеметричних даних. Симулятор дозволяє генерувати HTTP-запити, ідентичні тим, що надсилає мікроелектронний пристрій. Це забезпечує можливість тестування бази даних та інтерфейсів візуалізації без використання фізичного обладнання, що значно прискорює процес налагодження системи. Для імітації безперервного потоку телеметричних даних у режимі реального часу було розроблено програмний модуль, що використовує таймери на стороні клієнта (JavaScript setInterval). Це дозволяє протестувати пропускну здатність API-інтерфейсу та стабільність збереження даних у таблиці телеметрії під час тривалих спортивних сесій. Інтерфес продемонстровано на рисунку 2.28.

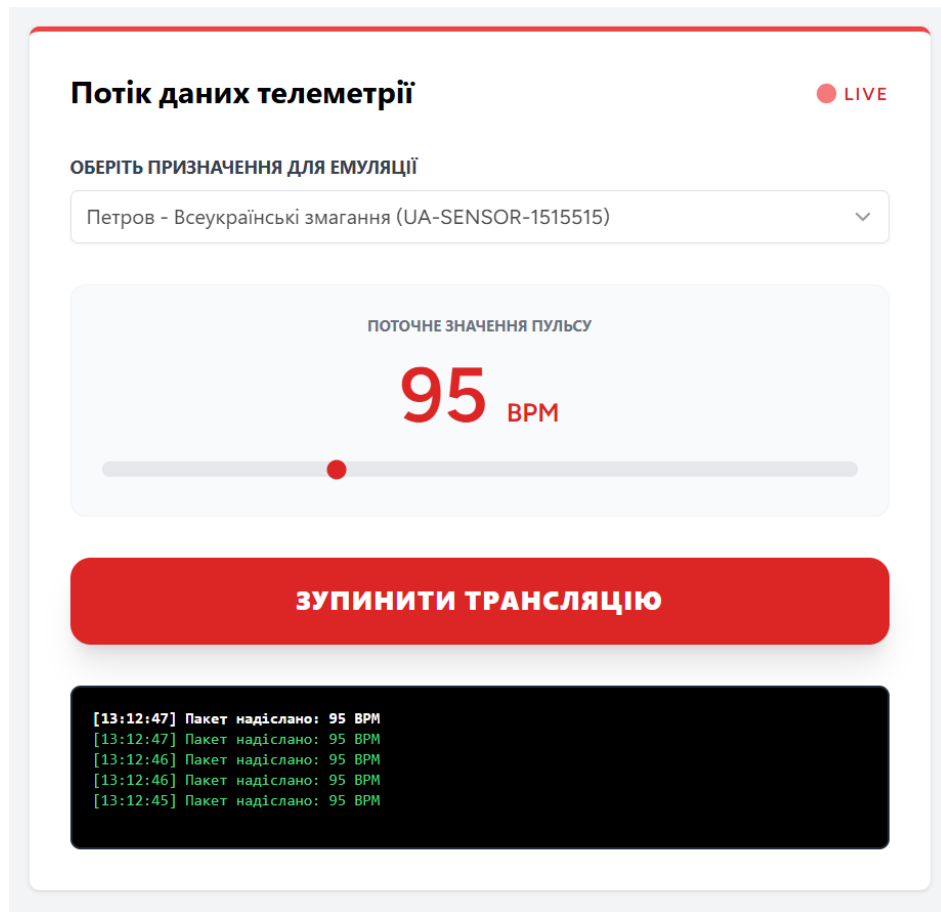


Рисунок 2.28 – Розділ симуляції передачі даних датчику

Заключним етапом розробки стало створення модуля візуалізації телеметричних даних у реальному часі. Для побудови графіків використано бібліотеку Chart.js, яка інтегрована у Vue-компоненти. Система реалізує принцип реактивного оновлення: кожні 500 мс клієнтська частина виконує запит до бази даних для отримання нових показників пульсу спортсмена. Отримані дані динамічно додаються до часової шкали графіка без перезавантаження сторінки. Доступ до моніторингу реалізовано як для адміністраторів, так і для неавторизованих користувачів (гостей), що дозволяє використовувати систему для публічної трансляції результатів змагань. Інтерфейс графіку – на рисунку 2.29.

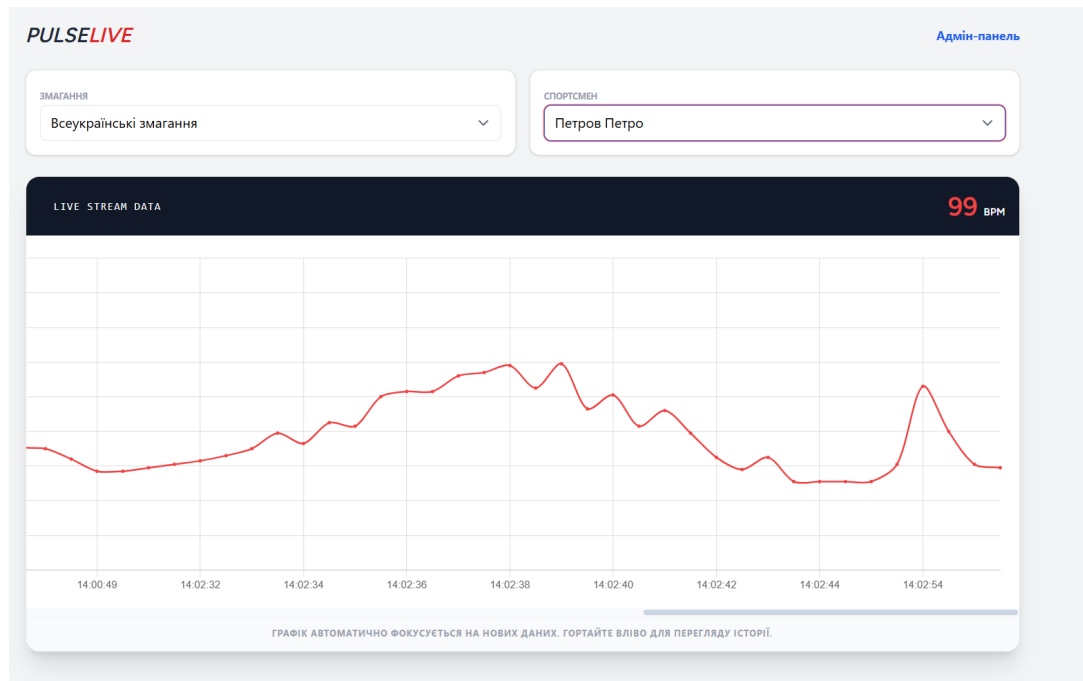


Рисунок 2.29 – Результати дослідження передачі параметрів пульсу спортсмена у реальному часі

У результаті проведеної роботи було спроектовано та реалізовано комплексну мікроелектронну інформаційну систему проведення спортивних змагань. Розроблена веб-платформа на базі фреймворку Laravel 10 та прогресивного JS-фреймворку Vue.js 3 забезпечує повний життєвий цикл управління спортивними заходами: від реєстрації учасників та апаратних засобів до візуалізації телеметричних даних у режимі реального часу.

Ключовою особливістю системи є інтегрований модуль моніторингу з використанням бібліотеки Chart.js, що дозволяє спостерігати за фізичним станом спортсменів (частотою серцевих скорочень) безпосередньо під час змагання. Реалізована архітектура з використанням REST API та механізмів інерційної передачі даних (Inertia.js) продемонструвала високу стабільність та низьку затримку при обробці поточних даних. Створений імітаційний модуль підтвердив готовність системи до інтеграції з реальними мікроконтролерами ESP32, що робить розроблене програмне забезпечення ефективним інструментом для цифровізації сучасного спортивного процесу.

3. ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ

У сучасній спортивній медицині та інженерії розрізняють кілька підходів до побудови систем реєстрації ЧСС.

Перший сегмент представлений професійними комерційними рішеннями (типу Polar H10). Це пристрої з еталонною точністю, що використовують золотий стандарт ЕКГ-вимірювань. Проте вони мають закриту архітектуру, що ускладнює їх пряму інтеграцію з кастомними WEB-платформами без використання дорогих ліцензійних SDK.

Другий напрямок – аматорські системи на базі універсальних плат (Arduino-сумісні модулі). Вони вирізняються низькою ціною та доступністю компонентів, але мають критичні недоліки: значні габарити, високе енергоспоживання та відсутність підтримки сучасних протоколів великої дальності, що робить їх непридатними для професійного спорту.

Третій варіант – напівпрофесійні датчики середнього цінового сегменту. Вони забезпечують базову точність, проте часто використовують застарілі версії Bluetooth (4.0/4.2), що обмежує радіус дії системи умовами невеликого закритого приміщення.

Розроблена у даному проєкті система базується на поєднанні спеціалізованого медичного аналогового фронтенду та сучасного енергоефективного мікроконтролера з підтримкою BLE 5.0 Long Range. Це дозволяє отримати точність професійного рівня при повній відкритості програмного стеку для інтеграції з Laravel-бекендом та суттєвому зниженні собівартості виробництва.

Для вибору оптимального апаратного рішення в межах даного проєкту необхідно провести багатокритеріальний аналіз існуючих технологій зняття біометричних даних. З цією метою сформовано ієрархію чотирьох альтернативних варіантів (табл. 3.1).

Таблиця 3.1 – Можливі варіанти реалізації вимірювального вузла

Позначення	Тип рішення	Короткий опис
A	Комерційний датчик преміум-класу (Polar H10)	Висока точність, закрита екосистема, висока ціна.
B	Аматорська система (Arduino + PulseSensor)	Мінімальна ціна, низька стабільність, малий радіус дії.
C	Комерційний датчик середнього сегменту	Прийнятна ціна, обмежені можливості Long Range.
D	Розроблена система (nRF52840 + AD8232)	Власна архітектура, підтримка BLE 5.0 Long Range, пряма інтеграція з web-системою.

Для вибору оптимальної архітектури вимірювального вузла використано метод аналізу ієрархій. Порівняння проводиться за чотирма критеріями: інтегрованість у WEB-систему (1), дальність зв'язку (2), надійність реєстрації даних (3) та собівартість (4).

Для визначення вагомості кожного критерію використовується стандартна шкала відносної важливості (табл. 3.2).

Таблиця 3.2 – Шкала відносної важливості

Інтенсивність відносної важливості	Визначення
1	рівна важливість
3	помірна перевага
5	сильна перевага
7	значна перевага
9	дуже сильна перевага
2,4,6,8	проміжні судження

Розрахунок локальних пріоритетів X_i для кожного критерію здійснюється за формулою (3.1):

$$X_i = \frac{\sqrt[n]{\prod_{j=1}^n \omega_{ij}}}{\sum_{i=1}^n \sqrt[n]{\prod_{j=1}^n \omega_{ij}}} \quad (3.1)$$

де n – кількість критеріїв ($n=4$);

ω_{ij} – ступінь переваги i -го критерію над j -м (згідно зі шкалою відносної важливості);

$\prod_{j=1}^n \omega_{ij}$ – добуток елементів у рядку матриці попарних порівнянь;

$\sum_{i=1}^n (\dots)$ – сума середніх геометричних значень усіх рядків матриці.

На основі аналізу специфіки проєкту, де основною вимогою є створення цілісної системи (Hardware + Web), було сформовано матрицю попарних порівнянь критеріїв (табл. 3.3). При заповненні матриці прийнято наступну логіку експертних оцінок:

Інтегрованість має помірну перевагу (3) над дальністю, оскільки без можливості автоматичного прийому даних веб-системою проєкт втрачає свою функціональну цінність.

Інтегрованість має сильну перевагу (5) над собівартістю, тому що для науково-дослідної роботи технічна досконалість та автоматизація є пріоритетнішими за економію коштів.

Дальність зв'язку має помірну перевагу (3) над надійністю, оскільки в умовах стадіону саме радіус дії радіоканалу є найбільш критичним вузьким місцем.

Всі зворотні значення ($1/3$, $1/5$ тощо) відображають відповідно меншу вагу одного критерію порівняно з іншим.

Таблиця 3.3 – Матриця попарних порівнянь критеріїв для системи моніторингу

Критерій	1	2	3	4	$\sqrt[n]{\prod_{j=1}^n \omega_{ij}}$	X_i
1. Інтегрованість	1	2	3	5	2,340	0,46
2. Дальність зв'язку	1/2	1	3	4	1,565	0,31
3. Надійність	1/3	1/3	1	3	0,760	0,15
4. Собівартість	1/5	1/4	1/3	1	0,359	0,08
Σ					5,024	1

Найбільшу вагу отримав критерій інтегрованості ($X_1 = 0,46$), оскільки основною вимогою до роботи є створення цілісної системи з автоматичною передачею даних у розроблений WEB-інтерфейс.

Для обчислення глобального пріоритету розробленої системи (варіант D) у порівнянні з аналогами (A – Polar, B – DIY-модулі, C – Coospro), складено матриці порівняння альтернатив за найбільш вагомими критеріями.

Таблиця 3.4 – Порівняння альтернатив за критерієм «Інтегрованість»

Технологія	A	B	C	D	$\sqrt[4]{\prod_{i=1}^4 \omega_i}$	X_i
A	1	3	2	1/5	1,047	0,166
B	1/3	1	1/2	1/7	0,385	0,061
C	1/2	2	1	1/6	0,639	0,101
D	5	7	6	1	3,807	0,672
Σ					5,878	1

Таблиця 3.5 – Порівняння альтернатив за критерієм «Дальність зв'язку»

Технологія	A	B	C	D	$\sqrt[n]{\prod_{j=1}^n \omega_{ij}}$	X_i
A	1	5	3	1/3	1,495	0,263
B	1/5	1	1/4	1/8	0,281	0,049
C	1/3	4	1	1/5	0,719	0,126
D	3	8	5	1	3,310	0,562
Σ					5,805	1

Таблиця 3.6 – Порівняння альтернатив за критерієм «Надійність»

Технологія	A	B	C	D	$\sqrt[n]{\prod_{j=1}^n \omega_{ij}}$	X_i
A	1	7	4	3	3,027	0,553
B	1/7	1	1/3	1/5	0,312	0,057
C	1/4	3	1	1/2	0,782	0,143
D	1/3	5	2	1	1,351	0,247
Σ					5,472	1

Таблиця 3.7 – Порівняння альтернатив за критерієм «Собівартість»

Технологія	A	B	C	D	$\sqrt[n]{\prod_{j=1}^n \omega_{ij}}$	X_i
A	1	1/9	1/4	1/5	0,273	0,051
B	9	1	3	2	2,711	0,507
C	4	1/3	1	1/1,5	0,970	0,181
D	5	1/2	1,5	1	1,392	0,260
Σ					5,346	1

Підсумковий етап аналізу полягає в обчисленні глобального пріоритету (G_p) для кожної з розглянутих альтернатив. Глобальний пріоритет визначається як сума добутків локальних пріоритетів альтернатив на вагові коефіцієнти відповідних критеріїв за формулою (3.2):

$$G_p = \sum_{n=1}^4 (X_{crit_n} \times X_{alt_i_n}) \quad (3.2)$$

де X_{crit_n} – вага n-го критерію (з таблиці 3.3),

$X_{alt_i_n}$ – локальний пріоритет i-ої альтернативи за n-им критерієм (табл. 3.4-3.7)

Узагальнені результати розрахунку глобальних пріоритетів наведені у таблиці 3.8.

Таблиця 3.8 – Глобальний пріоритет альтернативних рішень

Пріоритети	№1	№2	№3	№4	Глобальний пріоритет
Вага	0,46	0,31	0,15	0,08	
A	0,166	0,263	0,553	0,051	0,245
B	0,061	0,049	0,057	0,507	0,092
C	0,101	0,126	0,143	0,181	0,121
D	0,672	0,562	0,247	0,260	0,541

Аналіз отриманих результатів (табл. 3.8) свідчить про те, що найбільш доцільним варіантом для реалізації у межах даного проєкту є варіант D – власна розробка мікроелектронної системи.

Незважаючи на те, що професійне рішення (варіант A) має найвищу оцінку за критерієм надійності, розроблений проєкт суттєво переважає аналоги за сукупністю факторів інтегрованості та дальності зв'язку. Високе значення глобального пріоритету розробленої системи (0,541) обумовлене використанням сучасного стеку Bluetooth 5.0 Long Range та можливістю безпосередньої передачі даних у Laravel-бекенд без посередництва закритих хмарних сервісів. Це підтверджує наукову та практичну обґрунтованість обраного напрямку розробки.

Для оцінки фінансових витрат проведено детальний розрахунок вартості компонентів для двох основних вузлів системи: нагрудного датчика та приймального шлюзу.

Розрахунок собівартості нагрудного датчика наведено у таблиці 3.9. При розрахунку враховано кількість елементів, необхідних для повноцінної об'язки АЦП та радіочастотного тракту.

Таблиця 3.9 – Вартість компонентів одного нагрудного датчика

Найменування компонента	К-сть	Ціна за од., грн	Сума, грн
Мікроконтролер nRF52840-QIAA	1	450.00	450.00
Підсилювач ЕКГ AD8232	1	185.00	185.00
Керамічна патч-антена SWLP.2450	1	85.00	85.00
Кварцовий резонатор 32 МГц	1	30.00	30.00
Резистори С2-33 (Україна)	12	1.80	21.60
Конденсатори К10-17 (Україна)	18	2.50	45.00
Світлодіод індикації КТ-0603	1	4.50	4.50
Перемикач живлення SWSPDT	1	20.00	20.00
Тримач батареї CR2032-BS-6-1	1	35.00	35.00
Елемент живлення CR2032	1	45.00	45.00
Друкована плата (PCB), виготовлення	1	120.00	120.00
Корпус (ABS-пластик, 3D-друк)	1	100.00	100.00
Разом собівартість датчика			1141.10

Розрахунок собівартості приймального шлюзу наведений у таблиці 3.10. Шлюз базується на готовому модулі, що зменшує кількість дискретних компонентів.

Таблиця 3.10 – Вартість компонентів приймального шлюзу

Найменування компонента	К-сть	Ціна за од., грн	Сума, грн
Модуль ESP32-S3-WROOM-1-N8	1	260.00	260.00
Регулятор напруги AMS1117-3.3	1	18.00	18.00
Конденсатори фільтрації (100 мкФ)	2	12.00	24.00
Резистор (LED)	1	1.80	1.80
Світлодіод індикації	1	4.50	4.50
Роз'єм USB-C (живлення)	1	45.00	45.00
Друкована плата шлюзу	1	130.00	130.00
Корпус шлюзу	1	120.00	120.00
Разом собівартість шлюзу			603.30

Тепер оцінимо загальну вартість системи та економічний ефект. Для об'єктивного порівняння вартості розгортання системи на змаганнях з $N = 50$ учасниками розглянемо два сценарії. В обох випадках для покриття зони змагань (стадіону) та ретрансляції даних у глобальну мережу необхідно використовувати мінімум 2 приймальні шлюзи.

Обрахуємо витрати за формулою 3.3:

$$E = (N \times C_{\text{sensor}}) + 2 \times C_{\text{gateway}} \quad (3.3)$$

де C_{sensor} – вартість датчика, C_{gateway} – вартість шлюзу.

Отже, для системи, розробленої в рамках проєкту, $E_{\text{project}} = (50 \times 1141,10) + 2 \times 603,30 = 58261,60$ грн.

Порахуємо те саме, але для комерційних датчиків (для прикладу Polar H10) – $E_{\text{commercial}} = (50 \times 4500) + 2 \times 603,30 = 226206,60$ грн.

Для наочної оцінки переваг розробленого рішення розрахуємо відносну економію коштів ($E\%$), що досягається при впровадженні власної мікроелектронної системи порівняно із закупівлею комерційних аналогів, за формулою 3.4:

$$E_{\%} = \frac{E_{commercial} - E_{project}}{E_{commercial}} \times 100\% \quad (3.4)$$

Підставивши значення з попередніх розрахунків, отримаємо:

$$E_{\%} = \frac{226206,60 - 58261,60}{226206,60} \times 100\% \approx 74.24\%$$

Впровадження розробленого апаратно-програмного комплексу дозволяє знизити витрати на технічне забезпечення спортивних змагань на 74,24%. Це означає, що за ціною одного комплекту обладнання на базі Polar H10 можна розгорнути майже чотири аналогічні системи на базі розроблених модулів nRF52840 та ESP32-S3.

Додатково слід врахувати, що власне рішення на базі nRF52840 підтримує режим Coded PHY (Long Range), що дозволяє використовувати меншу кількість шлюзів на одиницю площі стадіону порівняно з комерційними датчиками Polar, які зазвичай обмежені стандартним радіусом дії BLE (10-15 м), що у реальних умовах могло б ще більше збільшити розрив у вартості на користь розробленого проєкту.

Такий високий показник економічної ефективності в поєднанні з покращеними характеристиками дальності зв'язку робить проєкт комерційно привабливим для спортивних федерацій та організаторів масових змагань, що потребують масштабованих рішень при обмеженому бюджеті.

4. ОХОРОНА ПРАЦІ ТА ТЕХНОГЕННА БЕЗПЕКА

4.1 Електробезпека при роботі з електроприладами та офісною оргтехнікою

Охорона праці при роботі з електроприладами та офісною оргтехнікою – комплекс заходів для створення безпечних умов при роботі з електричним устаткуванням, обладнанням, силовими кабелями та іншою технікою. Неправильне застосування цих приладів, або неакуратне поводження з ними може призвести до ураження струмом, пожежі на підприємстві і т.д. Працівник зобов'язаний дбати про особисту безпеку і здоров'я, а також про безпеку і здоров'я оточуючих людей в процесі виконання будь-яких робіт чи під час перебування на території підприємства. Він повинен знати і виконувати вимоги нормативно-правових актів з охорони праці, правила поводження з машинами, механізмами, устаткуванням та іншими засобами виробництва, користуватися засобами колективного та індивідуального захисту.

Кожен працівник повинен пройти інструктаж з електробезпеки та інструктаж з охорони праці при користуванні електро побутовими приладами та офісною оргтехнікою. Крім того, він має ознайомитись з інструкцією з експлуатації використовуваного приладу, що надається виробником.

Користування електро побутовими приладами та офісною оргтехнікою вимагає особливої уваги та обережності від людини. Щоб приступати до роботи з такою технікою, потрібно знати принципи її роботи та правильної експлуатації; знати порядок підключення та відключення (у тому числі у

екстрених ситуаціях, що загрожують життю та здоров'ю працівника та оточуючих).

Основні види небезпеки, які можуть впливати на користувача електроприладами:

- ураження організму електрострумом;
- загроза термоопіків;
- канцерогенні речовини;
- лазерний промінь;
- підвищення в повітрі вмісту озону, оксидів азоту і вуглецю;
- шум від працюючого обладнання.

Для зниження вірогідності позаштатних ситуацій, які загрожують життю та здоров'ю працівників підприємства, кабелі електро побутових приладів та офісної оргтехніки повинні розташовуватися так, щоб їх не можна було пошкодити необережним рухом.

Перед початком роботи з електроприладами та офісною оргтехнікою користувач повинен провести візуальний огляд електро побутових приладів та офісної оргтехніки, переконатися у справності електророзеток, штепсельних вилок до електромережі. Також необхідно перевірити:

- клас напруги приладу та розетки, до якої підключається прилад;
- справність розетки, до якої буде підключатись електроприлад;
- наявність заземлення розетки;
- справність електропроводки приладу (вилки, шнура, місця їхнього з'єднання, відсутність пошкодження ізоляції проводу);
- відсутність зовнішніх механічних пошкоджень приладу;
- відсутність поблизу легкозаймистих речовин.

Відстань між електроприладами і стінами повинні витримуватись згідно із «Інструкціями з експлуатації електроприладів», наданих заводом виробником, правилами пожежної безпеки. Відстань повинна давати змогу безперешкодного вимкнення приладу від електромережі у разі необхідності.

По закінченні робочого дня вимкнути вимикач на електропобутових приладах та офісної оргтехніки та від'єднати провід живлення від розетки електромережі. При цьому слід пам'ятати, що від'єднуючи вилку електропобутових приладів та офісної оргтехніки від розетки її слід тримати за корпус, а не смикати за провід живлення, бо можна висмикнути один з проводів і потрапити під дію електричного струму. [18]

4.2 Безпека при роботі з датчиками пульсу

Техніка безпеки при роботі з пульсометрами включає в себе декілька важливих аспектів, щоб уникнути травм та забезпечити точні вимірювання. Необхідно правильно використовувати пристрій, стежити за його станом та дотримуватись інструкцій виробника.

Основні правила безпеки при роботі з пульсометрами:

- Перевірка пристрою: перед кожним використанням необхідно перевірити пульсометр на відсутність пошкоджень, таких як тріщини на корпусі або несправності ремінця. Не використовуйте пошкоджений пристрій.
- Правильне встановлення: переконайтеся, що пульсометр правильно закріплений на тілі, щоб уникнути зміщень під час руху. Не затягуйте ремінець надмірно, щоб уникнути дискомфорту або порушення кровообігу.
- Захист від вологи: пульсометри можуть бути чутливі до вологи. Уникайте контакту з водою, якщо пристрій не є водонепроникним.
- Захист від екстремальних температур: уникайте впливу на пульсометр екстремально високих або низьких температур, оскільки це може вплинути на його роботу та точність вимірювань.
- Захист від ударів: уникайте падінь та ударів пульсометра, які можуть пошкодити його електронні компоненти.

- Дотримання інструкцій виробника: завжди читайте та дотримуйтесь інструкцій виробника щодо використання та догляду за пульсометром.
- Використання за призначенням: використовуйте пульсометр виключно за його прямим призначенням – для вимірювання пульсу.
- Заміна батарейок: за потреби замінюйте батарейки у відповідності до інструкцій виробника.
- Зберігання: зберігайте пульсометр у сухому та безпечному місці, недоступному для дітей та домашніх тварин.
- Очищення: регулярно очищуйте пульсометр від бруду та поту, використовуючи м'яку тканину.

У деяких випадках, наприклад, під час інтенсивних тренувань, може бути доцільним використання додаткових засобів захисту, таких як спортивні пов'язки на зап'ястя, щоб уникнути подразнень шкіри.

Дотримання цих правил допоможе вам безпечно та ефективно використовувати пульсометр, отримуючи точні дані про ваш пульс [19].

Найважливішим аспектом безпеки при використанні нагрудних датчиків є забезпечення мінімального струму витоку через тіло спортсмена. У розробленій системі це реалізовано шляхом встановлення послідовних обмежувальних резисторів номіналом 180 кОм у кожному входному каналі електродів. Згідно з розрахунками за законом Ома, навіть у разі критичного пробоя системи живлення та потрапляння напруги 3.3 В на контактну групу, струм через грудну клітку не перевищить 18.3 мкА. Це відповідає жорстким вимогам стандарту IEC 60601-1 для виробів типу CF [16], де максимально допустимий струм витоку в нормальному стані становить 10 мкА, а в умовах одиничного пошкодження – до 50 мкА. Таким чином, апаратна архітектура пристрою гарантує відсутність ризику фібриляції або подразнення серцевого м'яза навіть при тривалій експлуатації в умовах підвищеної вологості (потовиділення).

4.3 Пожежна безпека та експлуатація автономних джерел живлення

Енергозабезпечення системи базується на використанні літієвих елементів живлення: CR2032 (у датчику) та Li-Po акумуляторів (у шлюзі). Основним ризиком для літій-полімерних батарей є термічний розгін у разі короткого замикання або перегріву. Для нівелювання цієї загрози у шлюзі на ESP32-S3 застосовано схему стабілізації на базі регулятора AMS1117-3.3, який має вбудований захист від перевищення струму та теплове відключення. Для нагрудного датчика використання елемента CR2032 є безпечним через його високий внутрішній опір, що фізично обмежує струм короткого замикання до безпечних значень. Рекомендується дотримання правил утилізації згідно з Директивою 2006/66/EC, що виключає потрапляння важких металів у навколишнє середовище та запобігає випадковому займанню відпрацьованих елементів. [20]

ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ

1. У ході аналізу можливостей використання вітчизняної елементної бази для створення комплексної мікроелектронної системи моніторингу біометричних даних під час спортивних змагань, встановлено, що вона забезпечить стабільний потік телеметричних даних та дасть можливість локалізації виробництва таких пристроїв в Україні.

2. Розроблена принципова електрична схема нагрудного датчика пульсу на спеціалізованому аналоговому фронтенді AD8232 та системі на чіпі nRF52840, при цьому використання захисного резистивно-конденсаторного бар'єру забезпечить повну електробезпеку спортсмена та завадостійкість вимірювань.

3. Проведено дослідження та впровадження сучасних радіочастотних технологій, зокрема стандарту Bluetooth 5.0 Long Range, що дозволить збільшити радіус стабільного збору даних на відкритих стадіонах до 200–250 метрів без підвищення енергоспоживання, що є суттєвою перевагою над комерційними аналогами.

4. Розроблено інтелектуальний приймальний шлюз на базі модуля ESP32-S3, що дозволить реалізувати двоядерну обробку даних, забезпечивши паралельне обслуговування радіочастотного стеку та мережевого з'єднання, а програмне забезпечення успішно виконає трансформацію бінарних пакетів у формат JSON та їх ретрансляцію на WEB-сервер за протоколом HTTP POST.

5. Рекомендовано для покриття надвеликих локацій подальше масштабування мережі через розгортання ретрансляторів, об'єднаних за топологією Mesh-мережі, а також розширення веб-системи інтелектуальним

модулем на базі алгоритмів машинного навчання, що відкриває можливості для автоматичної детекції ознак перевтоми спортсменів, що зробить систему незамінним інструментом для професійного тренерського штабу.

ПЕРЕЛІК ПОСИЛАНЬ

1. Березанська М.О. Огляд та сучасний стан датчиків серцевого ритму : стаття. Київ : КПІ ім. Ігоря Сікорського, 2019. 4 с.
2. Н.Г. Іванушкіна, К.О. Іванько, А.О. Попов, Є.С.Карплюк, О.П.Шуляк. Біомедичні електронні системи : Конспект лекцій. – Київ : КПІ ім. Ігоря Сікорського, 2020. 113 с.
3. Polar H10 : Polar Electro. URL: <https://www.polar.com/ua/sensors/h10-heart-rate-sensor> (дата звернення: 17.06.2025).
4. Garmin HRM-Pro Plus : Heart Rate Monitor. URL: <https://www.garmin.com/en-US/p/770963/pn/010-13118-05/> (дата звернення 17.06.2025).
5. Wahoo TICKR X : TRACKR Heart Rate Monitor Chest Strap. URL: <https://eu.wahoofitness.com/devices/heart-rate-monitors/trackr-heart-rate-buy> (дата звернення: 17.06.2025).
6. Волканін Є.Є. Застосування мікроконтролерів : текст лекції. Кременчук : Кременчуцький льотний коледж, 2023. 19 с.
7. Грищук Ю.С. Мікропроцесорні пристрої : навчальний посібник. – Харків : НТУ «ХПІ», 2007. 280 с.
8. Чешко І. В. Вступ до спеціальності «Електроніка» : навчальний посібник – Суми : Сумський державний університет, 2017. 148 с.
9. Arduino Nano 33 IoT – Secure 32-bit IoT Board with WiFi&BLE – Arduino Official Store. URL: <https://store.arduino.cc/products/arduino-nano-33-iot> (дата звернення: 1.07.2025).

10. Pico-series Microcontrollers – Raspberry Pi Documentation. URL: <https://www.raspberrypi.com/documentation/microcontrollers/pico-series.html> (дата звернення: 1.07.2025).
11. Espressif Centralized Documentation. URL: https://documentation.espressif.com/esp32-wroom-32_datasheet_en.pdf (дата звернення: 1.07.2025).
12. HTML Standard. URL: <https://html.spec.whatwg.org/> (дата звернення: 1.07.2025).
13. Cascading Style Sheets. URL: <https://www.w3.org/Style/CSS/> (дата звернення: 1.07.2025).
14. ECMA-262 – Ecma International. URL: <https://ecma-international.org/publications-and-standards/standards/ecma-262/> (дата звернення: 1.07.2025).
15. AD8232 Data Sheet. URL: <https://www.analog.com/media/en/technical-documentation/data-sheets/ad8232.pdf> (дата звернення: 1.08.2025).
16. IEC 60601-1-11:2015/Amd 1:2020. Medical electrical equipment – Part 1-11: General requirements for basic safety and essential performance – Collateral standard: Requirements for medical electrical equipment and medical electrical systems used in the home healthcare environment. URL: <https://www.iso.org/standard/72632.html> (дата звернення: 1.08.2025).
17. The Model View Controller: a Composed Pattern. Toni Sellarès, Universitat de Girona. URL: <https://ima.udg.edu/~sellares/EINF-ES1/MVC-Toni.pdf> (дата звернення: 3.05.2025).
18. Млавець Ю.Ю. Охорона праці : конспект лекцій для студентів математичного факультету і факультету післядипломної освіти та доуніверситетської підготовки – Ужгород : ДВНЗ “УжНУ”, 2015. 56 с.
19. Гриненко Є. В. Правила безпеки при користуванні сучасним медичним обладнанням / Є. В. Гриненко // Безпека життя і діяльності людини: теорія та практика : збірник наук. праць Всеук. наук.-практ. конф.,

присвяченої Всесвітньому Дню цивільної оборони та Всесвітньому Дню охорони праці. (Полтава, 25–26 квітня 2019 р.) / упоряд., і ред. : В. П. Титаренко, А. М. Хлопов. – Полтава : ПНПУ, 2019. с. 328-332

20. Directive 2006/66/EC of the European Parliament and of the Council on batteries and accumulators and waste batteries. URL: <https://eur-lex.europa.eu/eli/dir/2006/66/oj/eng> (дата звернення: 15.09.2025)

ДОДАТОК А

Таблиця А.1 – Перелік елементів до схеми електричної принципової датчика пульсу

Познач.	Найменування	Кільк.	Примітка
	<i>Мікросхеми</i>		
U1	Інструментальний підсилювач AD8232ACPZ	1	Analog Devices
U2	Система на чіпі nRF52840-QIAA-R	1	Nordic Semi
	<i>Резистори</i>		
R1, R2	C2-33-0,062 Вт 180кОм $\pm 5\%$	2	
	<i>Конденсатори (K10-17)</i>		
C1, C2	100 нФ $\pm 10\%$	2	
C3	1 мкФ $\pm 10\%$	1	
	<i>Радіокомпоненти</i>		
X1	Кварцовий резонатор 32 МГц (пасивний)	1	
U3	Антенa керамічна патч SWLP.2450.12.4.B.02	1	Taoglas
	<i>Джерело живлення та комутація</i>		
B1	Тримач батареї CR2032-BS-6-1	1	
Bat 1	Літійовий елемент живлення CR2032 3V	1	
SW1	Перемикач мікрослайдний SWSPDT	1	
P1	Роз'єм штирьовий (Електроди) 2.54mm 1x2	1	

Таблиця А.2 – Перелік елементів до схеми електричної принципової приймального шлюзу

	<i>Мікросхеми та модулі</i>		
U1	Модуль ESP32-S3-WROOM-1-N8	1	Espressif
U2	Регулятор напруги AMS1117-3.3	1	LDO
	<i>Резистори</i>		
R1	C2-33-0,062 Вт 220 Ом $\pm 5\%$	1	
	<i>Конденсатори (K50-35 або аналог)</i>		
C1, C2	100 мкФ 16В (електролітичний)	2	
	<i>Індикація та порти</i>		
LED1	Світлодіод червоний KT-0603R	1	
J1	Роз'єм USB Type-C (Female)	1	Живлення 5В

ДОДАТОК Б

```

#include <bluefruit.h>

// Визначення сервісів BLE згідно зі стандартом Bluetooth SIG
BLEService    hrs = BLEService(UUID16_SVC_HEART_RATE);
BLECharacteristic    hrc                                     =
BLECharacteristic(UUID16_CHR_HEART_RATE_MEASUREMENT);
BLECharacteristic    hrm                                     =
BLECharacteristic(UUID16_CHR_BODY_SENSOR_LOCATION);

String DEVICE_ID;      // Унікальний апаратний ідентифікатор датчика
const int SENSOR_PIN = A0; // Аналоговий вхід (пін P0.02)

// Змінні для обчислення часових інтервалів між ударами серця
unsigned long lastBeatTime = 0;
int heartRate = 0;

void setup() {
    // Serial використовується для відлагодження (Debug) через USB
    Serial.begin(115200);

    // ГЕНЕРАЦІЯ УНІКАЛЬНОГО ID ПРИСТРОЮ
    // Читаємо заводські регістри FICR (Factory Information Configuration
Registers)
    uint32_t deviceAddrHigh = NRF_FICR->DEVICEADDR[1] | 0xC000;
    uint32_t deviceAddrLow  = NRF_FICR->DEVICEADDR[0];
    DEVICE_ID  =  "UA-SENSOR-"  +  String(deviceAddrHigh,  HEX)  +
String(deviceAddrLow, HEX);
    DEVICE_ID.toUpperCase();

```

```

// НАЛАШТУВАННЯ BLE
Bluefruit.begin();

// Встановлення потужності передавача (+4 dBm – максимальна для даного
чіпа)
Bluefruit.setTxPower(4);
Bluefruit.setName(DEVICE_ID.c_str());

// АКТИВАЦІЯ РЕЖИМУ LONG RANGE (CODED PHY)
// Це ключова команда для забезпечення дальності до 200 метрів
Bluefruit.configPrphPhy(BLE_GAP_PHY_CODED,
BLE_GAP_PHY_CODED);

// Ініціалізація стандартного сервісу Heart Rate Service
hrs.begin();

// Налаштування характеристики вимірювання (Notify дозволяє шлюзу
отримувати дані без запиту)
hrc.setProperties(CHR_PROPS_NOTIFY);
hrc.setPermission(SECMODE_OPEN, SECMODE_NO_ACCESS);
hrc.setFixedLen(2); // 1 байт прапорців + 1 байт значення BPM
hrc.begin();

// Налаштування локації датчика (Обов'язково для стандарту BLE HRM)
hrm.setProperties(CHR_PROPS_READ);
hrm.begin();
hrm.write8(1); // Код "1" відповідає локації Chest (Груди)

// НАЛАШТУВАННЯ РЕКЛАМНИХ ПАКЕТІВ (ADVERTISING)

```

```
Bluefruit.Advertising.addFlags(BLE_GAP_ADV_FLAGS_LE_ONLY_GENERAL_DISC_MODE);
```

```
Bluefruit.Advertising.addService(hrs);
```

```
Bluefruit.Advertising.addName(); // Передаємо "UA-SENSOR-XXXX" в ефір
```

```
// Автоматичний перезапуск реклами при розриві з'єднання
```

```
Bluefruit.Advertising.restartOnDisconnect(true);
```

```
Bluefruit.Advertising.start(0); // 0 = безстроково
```

```
}
```

```
void loop() {
```

```
// Зчитування аналогового сигналу з підсилювача AD8232
```

```
int sensorValue = analogRead(SENSOR_PIN);
```

```
// АЛГОРИТМ ДЕТЕКЦІЇ R-ЗУБЦЯ
```

```
// Поріг 700 одиниць (при 10-бітному АЦП) відповідає піку серцевого скорочення
```

```
if (sensorValue > 700) {
```

```
    unsigned long currentTime = millis();
```

```
    unsigned long duration = currentTime - lastBeatTime;
```

```
// Фільтрація артефактів (мінімальний інтервал 300мс = макс 200 уд/хв)
```

```
if (duration > 300) {
```

```
    heartRate = 60000 / duration;
```

```
    lastBeatTime = currentTime;
```

```
// ПІДГОТОВКА ПАКЕТУ ДЛЯ ПЕРЕДАЧІ
```

```
// Формат згідно з GATT специфікацією: [Flags, Heart Rate Value]
```

```
// 0b00000110 означає: формат значення – UINT8, контакт з сенсором
```

підтверджено

```
uint8_t hrmData[2] = { 0b00000110, (uint8_t)heartRate };

// Надсилання сповіщення підключеному шлюзу (ESP32)
hrc.notify(hrmData, 2);

// Вивід у серійний порт для візуального контролю при налагодженні
Serial.print("Sensor ID: " + DEVICE_ID);
Serial.print(" | Heart Rate: ");
Serial.println(heartRate);
}
}

// Затримка 10мс для стабілізації роботи АЦП та зниження енергоспоживання
delay(10);
}
```

ДОДАТОК В

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <ArduinoJson.h>
#include <NimBLEDevice.h> // Покращена бібліотека для роботи з BLE

// Налаштування Wi-Fi
const char* ssid = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";

// Адреса вашої WEB-системи
const char* serverUrl = "https://sports-system.ua/api/receive_data ";

// Піни
const int LED_PIN = 1;

// UUID стандартного сервісу пульсу
static BLEUUID serviceUUID("180d");
static BLEUUID charUUID("2a37");

bool doConnect = false;
bool connected = false;
NimBLEAdvertisedDevice* advDevice;
NimBLERemoteCharacteristic* pRemoteCharacteristic;

// Функція індикації (миготіння при успішній передачі)
void blinkLED() {
    digitalWrite(LED_PIN, HIGH);
```

```

    delay(100);
    digitalWrite(LED_PIN, LOW);
}

// Callback для отримання даних від датчика
void notifyCallback(NimBLERemoteCharacteristic* pChar, uint8_t* pData, size_t
length, bool isNotify) {
    if (length >= 2) {
        int heartRate = pData[1]; // Другий байт пакету згідно зі стандартом
        String sensorID = advDevice->getName().c_str();

        Serial.printf("Received BPM: %d from %s\n", heartRate, sensorID.c_str());

        // Якщо є Wi-Fi – відправляємо дані на сервер
        if (WiFi.status() == WL_CONNECTED) {
            sendDataToServer(sensorID, heartRate);
            blinkLED();
        }
    }
}

// Функція відправки JSON на сервер
void sendDataToServer(String deviceID, int bpm) {
    HTTPClient http;
    http.begin(serverUrl);
    http.addHeader("Content-Type", "application/json");

    // Формування JSON
    StaticJsonDocument<200> doc;
    doc["device_id"] = deviceID;

```

```
doc["bpm"] = bpm;
doc["status"] = "online";
```

```
String jsonResponse;
serializeJson(doc, jsonResponse);
```

```
// POST запит
```

```
int httpResponseCode = http.POST(jsonResponse);
```

```
if (httpResponseCode > 0) {
  Serial.printf("Server Response: %d\n", httpResponseCode);
} else {
  Serial.print("Error on sending POST: ");
  Serial.println(httpResponseCode);
}
http.end();
}
```

```
// Логіка сканування BLE пристроїв
```

```
class MyAdvertisedDeviceCallbacks: public NimBLEAdvertisedDeviceCallbacks {
void onResult(NimBLEAdvertisedDevice* advertisedDevice) {
  // Шукаємо пристрій, ім'я якого починається на "UA-SENSOR-"
  if (advertisedDevice->getName().find("UA-SENSOR-") != std::string::npos) {
    Serial.print("Found UA Sensor: ");
    Serial.println(advertisedDevice->toString().c_str());

    NimBLEDevice::getScan()->stop();
    advDevice = advertisedDevice;
    doConnect = true;
  }
}
```

```

    }
};

void setup() {
    Serial.begin(115200);
    pinMode(LED_PIN, OUTPUT);

    // 1. Підключення до Wi-Fi
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("\nWi-Fi Connected!");

    // 2. Ініціалізація BLE
    NimBLEDevice::init("UA-Gateway-S3");

    // Активація Long Range (Coded PHY) для ESP32-S3
    NimBLEDevice::setCustomPhy(BLE_GAP_PHY_CODED,
    BLE_GAP_PHY_CODED, BLE_GAP_PHY_CODED);

    NimBLEScan* pNimBLEScan = NimBLEDevice::getScan();
    pNimBLEScan->setAdvertisedDeviceCallbacks(new
    MyAdvertisedDeviceCallbacks());
    pNimBLEScan->setActiveScan(true);
    pNimBLEScan->start(0, false);
}

void loop() {

```

```

// Якщо знайдено датчик – підключаємось
if (doConnect) {
    NimBLEClient* pClient = NimBLEDevice::createClient();
    if (pClient->connect(advDevice)) {
        NimBLERemoteService* pRemoteService = pClient-
>getService(serviceUUID);
        if (pRemoteService) {
            pRemoteCharacteristic = pRemoteService->getCharacteristic(charUUID);
            if (pRemoteCharacteristic && pRemoteCharacteristic->canNotify()) {
                pRemoteCharacteristic->subscribe(true, notifyCallback);
                connected = true;
            }
        }
    }
    doConnect = false;
}
delay(1000);
}

```

ДОДАТОК Г

```

namespace App\Http\Controllers\Api;
use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use App\Models\Sensor;
use App\Models\Assignment;
use App\Models\Telemetry;

class TelemetryController extends Controller
{
    public function store(Request $request)
    {
        // 1. Знаходимо датчик за його device_id
        $sensor = Sensor::where('device_id', $request->device_id)->first();
        if (!$sensor) return response()->json(['error' => 'Датчик не знайдено'], 404);
        // 2. Знаходимо активне призначення (останнє створене для цього датчика)
        $assignment = Assignment::where('sensor_id', $sensor->id)->latest()->first();
        if (!$assignment) return response()->json(['error' => 'Призначення не
знайдено'], 404);
        // 3. Записуємо телеметрію
        $data = Telemetry::create([
            'assignment_id' => $assignment->id,
            'bpm' => $request->bpm,
        ]);

        return response()->json(['status' => 'ok', 'id' => $data->id], 201);
    }
}

```

ДОДАТОК Д

```
namespace App\Http\Controllers;

use App\Models\Athlete;
use Illuminate\Http\Request;
use Inertia\Inertia;

class AthleteController extends Controller
{
    public function index()
    {
        return Inertia::render('Athletes/Index', [
            'athletes' => Athlete::all()
        ]);
    }

    public function store(Request $request)
    {
        $request->validate([
            'first_name' => 'required|string|max:255',
            'last_name' => 'required|string|max:255',
        ]);
        Athlete::create($request->all());
        return redirect()->back();
    }

    public function update(Request $request, $id)
    {

```

```
$request->validate([  
    'first_name' => 'required|string|max:255',  
    'last_name' => 'required|string|max:255',  
]);
```

```
  
$athlete = Athlete::findOrFail($id);  
$athlete->update($request->all());  
return redirect()->back();  
}
```

```
public function destroy($id)  
{  
    Athlete::findOrFail($id)->delete();  
    return redirect()->back();  
}  
}
```

ДОДАТОК Е

```
<script setup>

import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { useForm, Head, router } from '@inertiajs/vue3';

defineProps({
  assignments: Array,
  competitions: Array,
  athletes: Array,
  sensors: Array
});

const form = useForm({
  competition_id: "",
  athlete_id: "",
  sensor_id: "",
});

const submit = () => {
  form.post(route('assignments.store'), {
    onSuccess: () => form.reset(),
  });
};

const deleteAssignment = (id) => {
  if (confirm('Видалити призначення? Дані телеметрії для цього зв'язку можуть стати недоступними.')) {
    router.delete(route('assignments.destroy', id));
  }
}
```

```

};
</script>

<template>
  <Head title="Призначення датчиків" />
  <AuthenticatedLayout>
    <template #header>
      <h2 class="font-semibold text-xl text-gray-800 leading-tight">Призначення пристроїв</h2>
    </template>

    <div class="py-12">
      <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
        <div class="bg-white p-6 shadow sm:rounded-lg">

          <h3 class="text-lg font-bold mb-4">Нова прив'язка "Спортсмен - Датчик"</h3>

          <form @submit.prevent="submit" class="grid grid-cols-1 md:grid-cols-4 gap-4 mb-10 p-4 bg-orange-50 rounded border border-orange-100">
            <div>
              <label class="block text-xs font-bold text-orange-700">ЗМАГАННЯ</label>
              <select v-model="form.competition_id" class="w-full border-gray-300 rounded-md" required>
                <option value="" disabled>Оберіть змагання...</option>
                <option v-for="c in competitions" :key="c.id" :value="c.id">{{ c.title }}</option>
              </select>
            </div>

```

```

<div>
    <label class="block text-xs font-bold text-orange-700">СПОРТСМЕН</label>
    <select v-model="form.athlete_id" class="w-full border-gray-300 rounded-md" required>
        <option value="" disabled>Оберіть спортсмена...</option>
        <option v-for="a in athletes" :key="a.id" :value="a.id">{{
a.last_name }} {{ a.first_name }}</option>
    </select>
</div>
<div>
    <label class="block text-xs font-bold text-orange-700">ДАТЧИК
(DVICE ID)</label>
    <select v-model="form.sensor_id" class="w-full border-gray-300 rounded-md" required>
        <option value="" disabled>Оберіть пристрій...</option>
        <option v-for="s in sensors" :key="s.id" :value="s.id">{{
s.device_id }} ({{ s.model }})</option>
    </select>
</div>
<div class="flex items-end">
    <button type="submit" :disabled="form.processing" class="w-full bg-orange-600 text-white px-4 py-2 rounded-md hover:bg-orange-700 transition">
        Призначити
    </button>
</div>
</form>

<table class="w-full border-collapse">
    <thead>

```

```

<tr class="bg-gray-100 text-left">
  <th class="p-3 border">Змагання</th>
  <th class="p-3 border">Спортсмен</th>
  <th class="p-3 border">Датчик</th>
  <th class="p-3 border text-center">Дія</th>
</tr>
</thead>
<tbody>
<tr v-for="as in assignments" :key="as.id" class="hover:bg-gray-
50">
  <td class="p-3 border">{{ as.competition?.title }}</td>
  <td class="p-3 border font-bold">{{ as.athlete?.last_name }} {{
as.athlete?.first_name }}</td>
  <td class="p-3 border font-mono text-blue-600">{{
as.sensor?.device_id }}</td>
  <td class="p-3 border text-center">
    <button @click="deleteAssignment(as.id)" class="text-red-600
font-bold hover:underline">Видалити</button>
  </td>
</tr>
</tbody>
</table>
</div>
</div>
</div>
</AuthenticatedLayout>
</template>

```

ДОДАТОК Є

Шарапов Ілля

магістрант 1 курсу ІННІ ім. Ю. М. Потебні ЗНУ

Наук. кер.: к.т.н., проф. Ніконова З.А.

ІНФОРМАЦІЙНА WEB - СИСТЕМА ЗІ СПОРТИВНОЇ ТЕМАТИКИ НА ОСНОВІ MVC АРХІТЕКТУРИ

Спорт, як відомо, це вид діяльності, в якому тим чи іншим чином задіяні без перебільшення мільярди людей (спортсмени, параолімпійці, любителі, глядачі). Саме тому людство вже давно проводить спортивні змагання.

Для їх організації у сучасних умовах необхідним є застосування інформаційних систем. Вони використовуються для збереження, обробки результатів змагань; їх аналізу, класифікації; демонстрації широкому загалу людей; для безпосереднього автоматичного введення результатів змагань; тощо. На сьогодні Web-технології у інформаційних системах дозволяють оперувати результатами та іншими даними спортивних змагань через мережу Інтернет, з будь-якої точки планети зручним способом [1]. Основою систем є архітектура MVC, до якої входять:

- Model-View-Controller (MVC, «Модель-Представлення-Контролер») - схема поділу даних програми, призначеного для користувача інтерфейсу і керуючої логіки на три окремих компоненти: модель, уявлення і контролер - таким чином, що модифікація кожного компонента може здійснюватися незалежно. Можна виділити такі загальні ролі кожного компоненту:

- модель (Model) надає дані і реагує на команди контролера, змінюючи свій стан;
- представлення (View) відповідає за відображення даних моделі користувачеві, реагуючи на зміни моделі;
- контролер (Controller) інтерпретує дії користувача, сповіщуючи модель

про необхідність змін.

Отже, якщо відвідувач просто хоче отримати певну інформацію на сайті, його запит (в конкретному випадку – URL запити) аналізує контролер сайту. Якщо ресурс, на який робиться запит, не існує, то користувача буде переадресовано на сторінку, що повертає http-код 404. У випадку успіху запит передається спеціалізованому контролеру того чи іншого розділу сайту, який у свою чергу звертається до моделі за даними. Модель повертає дані з бази даних (БД), які потім передаються представленню для візуалізації інформації користувачеві.

Звичайно першочерговим етапом проєктування web-інтерфейсу є чітке визначення необхідних розділів та сторінок сайту. Інформаційна система повинна містити у собі 2 основні частини – гостьову та адміністративну. У свою чергу, кожний з цих розділів містить підрозділи. Автори вважають, що в рамках сучасного світу майже безальтернативною мовою розмітки, що використовується в сучасній Web-розробці, є HTML, яку і було використано в роботі. Це стандартизована мова розмітки документів у Всесвітній павутині. Більшість веб-сторінок містять опис розмітки на мові HTML (або XHTML). Мова HTML інтерпретується браузером; отриманий в результаті інтерпретації форматований текст відображається на екрані монітора комп'ютера або мобільного пристрою. Окрім HTML, було запропоновано ще CSS, яка також фактично є безальтернативною мовою опису зовнішнього вигляду документів, що були написані на мові розмітки HTML. Аналіз сучасних Web –технологій показав, що стосується мови для програмування клієнтської частини то з помітним відривом лідирує Javascript. Запропоновані програми, написані авторами на JavaScript, що виконуються в браузері користувача при завантаженні сторінки, дозволяють автоматично заповнювати форми, переформатовувати сторінки, приховувати небажаний вміст і вбудовувати бажаний, змінювати поведінку клієнтської частини веб-додатків, додавати елементи керування на сторінку. На сьогодні основними мовами Web-програмування у сфері back-end розробки є PHP, Java, C# та Python. Для

розробки веб-проєкту даного типу найдоцільніше було обрати мову PHP. Популярність PHP в області побудови веб-сайтів визначається наявністю великого набору вбудованих засобів для розробки веб-додатків. Основні з них:

- автоматичне вилучення POST і GET-параметрів, а також змінних оточення веб-сервера в визначені наперед масиви;
- взаємодія з великою кількістю різних систем управління базами даних;
- автоматизована відправка HTTP-заголовків;
- робота з HTTP-авторизацією;
- робота з cookies і сесіями;
- робота з локальними і віддаленими файлами, сокетам;
- обробка файлів, що завантажуються на сервер.

У світі сучасної веб-розробки використовуються різні реляційні системи управління базами даних. Серед лідерів – MySQL, PostgreSQL, Oracle, Microsoft SQL Server та інші. В якості системи управління БД в роботі було доцільно використати MySQL – безкоштовну та популярну, адже зв'язка PHP та MySQL є однією з найбільш уживаних в сфері веб-програмування. Автори вважають, що ця розробка дозволить сміливо використовувати цю систему для широкого загалу користувачів при проведенні спортивних змагань.

Література

1. Vermeij Arjan. A Generic MVC Model in Java. URL: <https://surli.cc/hapffi>
(дата звернення 25.03.2025 р.)

