

## КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини

Виконав(ла): студент(ка) IV курсу, групи СІ-41

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

Рій І.І.

(підпис)

(прізвище та ініціали)

Керівник

(підпис)

Паляниця Ю.Б.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Тиш Є.В.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Осухівська Г.М.

(прізвище та ініціали)

Рецензент

(підпис)

Стоянов Ю.М.

(прізвище та ініціали)

Міністерство освіти і науки України  
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії  
(повна назва факультету)

Кафедра комп'ютерних систем та мереж  
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Осухівська Г.М.

(підпис)

(прізвище та ініціали)

« 25 » 04 2024 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня бакалавр  
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»  
(шифр і назва спеціальності)

студенту Рію Ігорю Ігоровичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини

Керівник роботи Паляниця Юрій Богданович, к.т.н., ст. викладач  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 24 » 04 2024 року № 4/7-408

2. Термін подання студентом завершеної роботи \_\_\_\_\_

3. Вихідні дані до роботи Технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ

1. Аналіз технічного завдання

2. Проектна частина

3. Практична частина

4. Безпеки життєдіяльності, основи охорони праці

Висновки

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

Блок схеми роботи сайту

Блок схеми алгоритму роботи

Схема електрично принципова

Структурна схема

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Безпека життєдіяльності, основи охорони праці</i>	<i>Пилипець М.І., д.т.н., професор кафедри МТ</i>		

7. Дата видачі завдання \_\_\_\_\_

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Розробка та затвердження технічного завдання</i>	<i>01.02.24-04.02.24</i>	<i>Виконано</i>
2	<i>Аналіз можливих рішень поставленого завдання</i>	<i>05.02.24-11.02.24</i>	<i>Виконано</i>
3	<i>Розробка структурної, функціональної та електричної принципової схеми</i>	<i>06.05.24-11.05.24</i>	<i>Виконано</i>
4	<i>Вибір компонентів та програмного забезпечення для комп'ютеризованої системи</i>	<i>12.05.24-15.05.24</i>	<i>Виконано</i>
5	<i>Проектування апаратного забезпечення системи моніторингу біопказників</i>	<i>16.05.24-20.05.24</i>	<i>Виконано</i>
6	<i>Реалізація програмного забезпечення комп'ютеризованої системи</i>	<i>21.05.24-25.05.24</i>	<i>Виконано</i>
7	<i>Тестування системи моніторингу біопказників</i>	<i>26.05.24-30.05.24</i>	<i>Виконано</i>
8	<i>Оформлення кваліфікаційної роботи</i>	<i>01.06.24-13.06.24</i>	<i>Виконано</i>
9	<i>Попередній захист кваліфікаційної роботи</i>	<i>14.06.24</i>	<i>Виконано</i>
10	<i>Захист кваліфікаційної роботи</i>	<i>26.06.24</i>	

Студент

\_\_\_\_\_ (підпис)

Рій І.І.

\_\_\_\_\_ (прізвище та ініціали)

Керівник роботи

\_\_\_\_\_ (підпис)

Паляниця Ю.Б.

\_\_\_\_\_ (прізвище та ініціали)

## АНОТАЦІЯ

Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини// Кваліфікаційна робота бакалавра // Рій Ігор Ігорович// Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних систем та мереж, група СІ-41 // Тернопіль, 2024 // с. – 67, рис. – 29, табл. – 1, кресл. – 4, додат. – 3, бібліогр. – 19.

Ключові слова: Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини, NodeMcu, Max30102, мікроконтролер, електропровідність шкіри.

В кваліфікаційній роботі бакалавра розроблено комп'ютеризовану систему моніторингу біопоказників для тренування стресостійкості.

Кваліфікаційна робота складається з чотирьох розділів.

Перший розділ присвячений аналізу технічного завдання та перегляду можливих рішень завдання.

Другий розділ описує розробку системи та вибір інструментів для її реалізації.

В третьому розділі приводиться розробка алгоритму програмного забезпечення та реалізація проектних рішень. Також цей розділ включає тестування готової комп'ютерної системи.

Четвертий розділ описує безпеку життєдіяльності та основи охорони праці.

## ABSTRACT

Computerized system for monitoring biological indicators for training human stress resistance // Bachelor's thesis// Rii Ihor Ihorovich// Ternopil National Technical University named after Ivan Pulyu, Faculty of Computer and Information Systems and Software Engineering, Department of Computer Systems and Networks, Group CI-41 // Ternopil, 2024 // p. - 67 , fig. - 29 , tab. - 1 , sheets A1. - 4 , appendix. – 3, bibliography – 19 .

Keywords: computerized system for monitoring biological indicators for training human stress resistance, NodeMcu, MAX30102, microcontroller, galvanic skin respoce.

A computerized system for monitoring biological indicators for training human stress resistance was developed in the bachelor's qualification work.

The qualification work consists of four sections.

The first section is devoted to the analysis of the technical task and review of possible solutions to the task.

The second section describes the development of the system and the selection of tools for its implementation.

The third section describes the development of the software algorithm and the implementation of project solutions. Also, this section includes testing of finished computerized system.

The fourth chapter describes life safety and the basics of labor protection.

## ЗМІСТ

СПИСОК СКОРОЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ .....	10
1.1 Аналіз вимог до комп'ютерної системи.....	10
1.2 Основні вимоги до комп'ютеризованої системи .....	12
1.3 Аналіз можливих рішень поставленого завдання .....	14
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА.....	17
2.1 Розробка узагальненої структури комп'ютеризованої системи моніторингу біопказників для тренування стресостійкості .....	17
2.2 Обґрунтування вибору апаратного забезпечення комп'ютеризованої системи моніторингу біопказників для тренування стресостійкості .....	20
2.2.1 NodeMcuV3 esp8266.....	20
2.2.2 Датчик Groove Gsr.....	26
2.2.3 Датчик MAX30102 .....	29
2.2.4 Arduino IDE.....	30
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА .....	33
3.1 Реалізація проектних рішень .....	33
3.1.1 Опис функцій сайту.....	34
3.2 Створення формули для обчислення рівня стресу .....	35
3.3 Налаштування Arduino IDE для NodeMcu .....	36
3.5 Результат тестування.....	44
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ .....	46
4.1 Фізіологічний вплив факторів існування на життєдіяльність людини .....	46
4.2 Заходи щодо запобігання електронебезпеки .....	47
4.3 Вимоги до профілактичних медичних оглядів для працівників ПК .....	49
ВИСНОВКИ .....	51

					<i>КС КРБ 123.127.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розроб.</i>		Рій І.І.			<i>Комп'ютеризована система моніторингу біопказників для тренування стресостійкості</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушіє</i>
<i>Перевір.</i>		Паляниця Ю.Б.					5	67
<i>Реценз.</i>		Стоянов Ю.М.				<i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. контр.</i>		Тиш Є.В.						
<i>Затверд.</i>		Осухівська Г.М.						

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	52
Додаток А .....	54
Додаток Б.....	62
Додаток В .....	64

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

## СПИСОК СКОРОЧЕНЬ

Wi-Fi - Wireless Fidelity

GSR – Galvanic Skin Responce

HR - Heart Rate

IDE - Integrated Development Environment

I2C - Inter-Integrated Circuit

USB – Universal Serial Bus

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
						7
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		



## ВСТУП

У сучасному світі, що характеризується швидкими темпами життя та зростаючими вимогами до ефективності та продуктивності, проблема стресу стає все більш актуальною. Стрес впливає як на фізичний так і на психічний стан людини, зменшуючи її працездатність та якість життя. Одним із найбільш перспективних підходів до вирішення цієї проблеми є використання комп'ютеризованих систем моніторингу біопоказників, які дозволять точно визначати та аналізувати фізіологічні реакції організму на стресові ситуації.

Традиційні методи тренування стресостійкості, такі як фізичні вправи, медитація та психологічні техніки, мають свої переваги, але часто вони є ненадійними та не дають можливості отримати точну інформацію про стан організму. Комп'ютеризовані системи моніторингу біопоказників виключають ці недоліки, забезпечуючи детальне та постійне спостереження за різними фізіологічними параметрами. Це відкриває нові можливості для індивідуалізації підходів до тренування стресостійкості, адаптуючи методики під конкретні потреби та особливості кожної людини.

Сучасні технології збору та аналізу даних дозволяють фіксувати широкий спектр біопоказників, таких як частота серцевих скорочень, електропровідність шкіри рівень кисню в крові та інші. Використовуючи ці дані, можна створювати детальні профілі стресових реакцій та визначати найбільш ефективні методи для їх подолання. Такий підхід дозволяє не тільки підвищити ефективність тренувань, але й знизити ризик розвитку хронічних захворювань, пов'язаних зі стресом.

Комп'ютеризовані системи моніторингу біопоказників можуть бути корисними в різних сферах. Вони можуть використовуватися професійними спортсменами для оптимізації тренувального процесу, лікарями та психологами для лікування та профілактики стресових розладів, а також звичайними людьми для покращення загального стану здоров'я. Крім того, такі системи можуть

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
						8
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

сприяти науковим дослідженням у галузі психофізіології, надаючи нові дані про механізми стресу та його вплив на організм.

Розробка та впровадження таких систем потребує тісної співпраці між різними науковими дисциплінами, включаючи медицину, біологію та інженерію. Лише інтеграція знань та досвіду з різних галузей може забезпечити створення ефективних та надійних рішень. Таким чином, комп'ютеризовані системи моніторингу біопоказників для тренування стресостійкості людини є не лише інноваційним інструментом для покращення якості життя, але й важливим кроком у розвитку сучасної науки та технологій.

Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини є перспективним напрямом досліджень, що має великий потенціал для практичного застосування. Вона не тільки сприяє підвищенню ефективності методик тренування, але й відкриває нові можливості для наукових досліджень у галузі психофізіології та медицини. Системний підхід до розробки таких технологій є ключем до створення інноваційних рішень, які здатні відповідати викликам сучасного світу та сприяти здоров'ю і добробуту людини.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

## РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

### 1.1 Аналіз вимог до комп'ютерної системи

При розробці комп'ютеризованої системи моніторингу біопоказників для тренування стресостійкості важливо врахувати як функціональні, так і нефункціональні вимоги. Даний розділ аналізує ці вимоги для забезпечення оптимальної роботи системи та досягнення поставлених цілей.

Функціональні вимоги:

- збір даних біопоказників. Система повинна бути здатна реєструвати та зберігати біопоказники користувача, такі як рівень стресу. Для цього необхідно використовувати високоточні сенсори, що забезпечують точне та надійне вимірювання.

- аналіз даних. Необхідно забезпечити алгоритми для обробки та аналізу зібраних даних з метою виявлення тенденцій та патернів, що характеризують рівень стресу. Це може включати використання методів машинного навчання та статистичних моделей для прогнозування та інтерпретації стану користувача.

- візуалізація результатів. Система повинна забезпечувати інтуїтивно зрозумілий інтерфейс для відображення даних у вигляді графіків, таблиць або інших візуальних засобів, що допоможуть користувачеві зрозуміти свій стан. Важливо забезпечити можливість перегляду історії даних та порівняння показників у різні періоди часу.

- збереження конфіденційності. Система має відповідати вимогам конфіденційності та безпеки даних, зберігаючи всю інформацію користувачів у захищеному вигляді. Це включає використання шифрування для зберігання та передачі даних.

					<i>КС КРБ 123.127.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розроб.</i>		Рій І.І.			<i>РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевір.</i>		Паляниця Ю.Б.					10	67
<i>Реценз.</i>		Стоянов Ю.М.				<i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. контр.</i>		Тиш Є.В.						
<i>Затверд.</i>		Осухівська Г.М.						

Нефункціональні вимоги до системи включають кілька ключових аспектів. По-перше, система повинна забезпечувати високу продуктивність, тобто швидку обробку даних та миттєву реакцію на запити користувачів, що є критично важливим для аналізу та відображення результатів у реальному часі. Надійність системи також є пріоритетом, що вимагає уникнення збоїв у роботі та втрати даних. Це передбачає стійкість до помилок та наявність механізмів відновлення після збоїв. Масштабованість є ще однією важливою вимогою, що дозволяє системі обслуговувати велику кількість користувачів без втрати продуктивності, зокрема за рахунок використання хмарних технологій та розподілених обчислень. Крім того, система повинна бути сумісною з різними пристроями та операційними системами, такими як смартфони, планшети, ПК, а також підтримувати стандарти відкритих протоколів для інтеграції з іншими системами та сервісами. Важливо також забезпечити високий рівень юзабіліті, тобто інтуїтивно зрозумілий та зручний інтерфейс для користувачів різного рівня підготовки, враховуючи принципи UX/UI дизайну.

Вимоги до апаратного забезпечення також мають кілька важливих складових. Для збирання біопоказників необхідно використовувати високоточні сенсори, здатні вимірювати параметри з мінімальною похибкою, такі як частота серцевих скорочень, рівень кортизолу, варіабельність серцевого ритму та інші. Апаратна частина системи повинна мати достатній обсяг оперативної пам'яті та потужний процесор для забезпечення швидкої обробки даних, що є критично важливим для реального часу аналізу та виконання складних алгоритмів. Крім того, система повинна бути оснащена модулями для бездротового зв'язку (Wi-Fi) для передачі даних між сенсорами та основною обчислювальною платформою, що забезпечить зручність використання та мобільність системи. Автономні джерела живлення з тривалим часом роботи є необхідними для безперервного моніторингу біопоказників протягом тривалого часу. Останнім, але не менш важливим аспектом є захист від зовнішніх впливів, таких як волога, пил та механічні пошкодження, що забезпечить надійність та довговічність обладнання. На рис. 1.1 зображено приклад структури системи

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

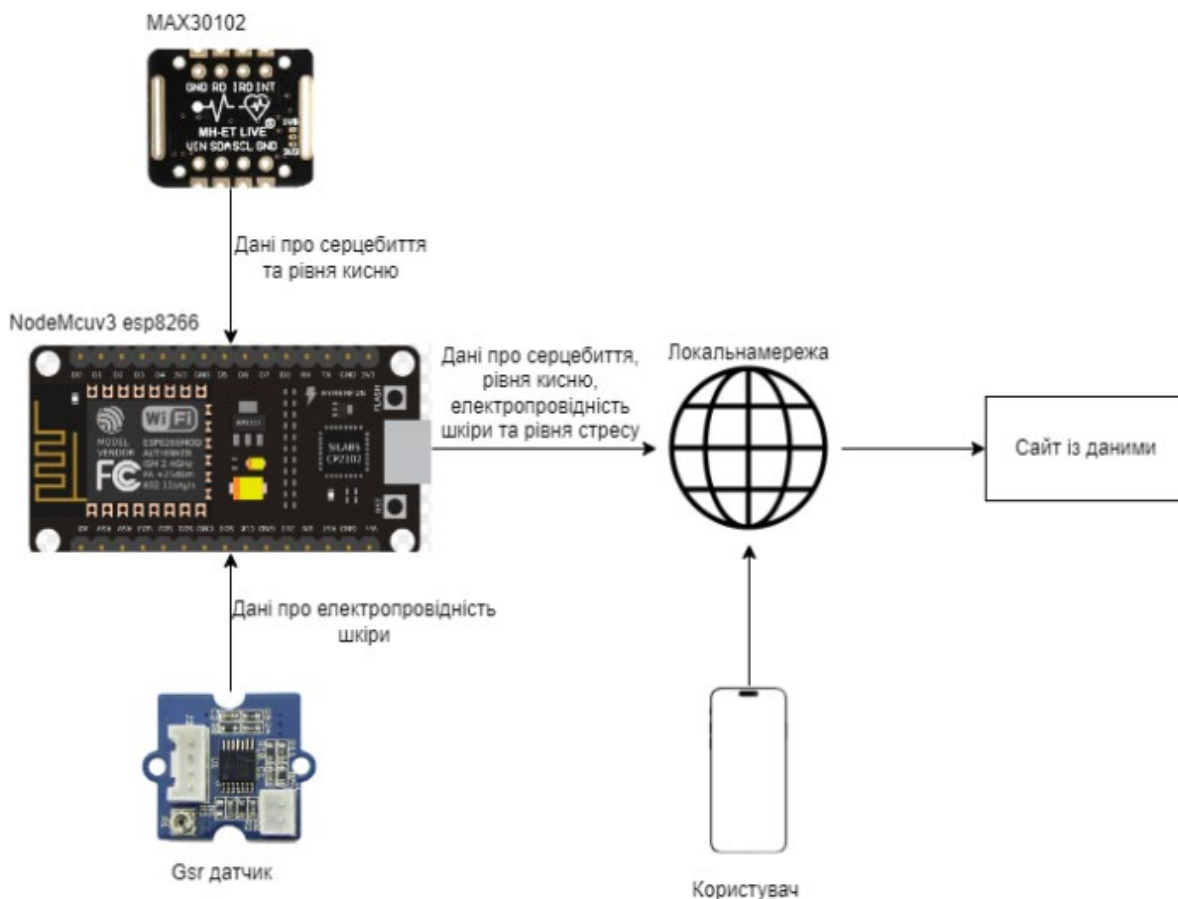


Рисунок 1.1 – Приклад структури системи

Ці вимоги допоможуть покращити систему та її функції для точнішого зчитування даних та обрахування.

## 1.2 Основні вимоги до комп'ютеризованої системи

Комп'ютеризована система моніторингу біопозначників повинна забезпечувати збір, аналіз та відображення даних у реальному часі. Вона повинна інтегрувати різноманітні сенсори для вимірювання ключових фізіологічних параметрів, таких як серцевий ритм, рівень кисню в крові, електрична активність шкіри та інші показники, що мають прямий або опосередкований зв'язок зі станом стресу. Система повинна вміти зберігати дані для подальшого аналізу,

Змн.	Арк.	№ докум.	Підпис	Дата

надавати можливість візуалізації динаміки показників, а також формувати звіти про стан користувача.

Програмне забезпечення має бути розроблена з урахуванням зручності користування та інтуїтивно зрозумілого інтерфейсу. Воно повинно включати модулі для збору даних з сенсорів, обробки та аналізу отриманих даних, а також візуалізації результатів. Особлива увага приділяється точності алгоритмів обробки даних, щоб забезпечити надійні результати. Крім того, програмне забезпечення повинно бути сумісним з різними операційними системами та мати можливість інтеграції з іншими програмними продуктами, наприклад, для подальшого аналізу даних у спеціалізованих статистичних пакетах.

Для забезпечення надійного функціонування системи необхідно використовувати високоточні та надійні сенсори, що гарантують точність вимірювань біопоказників. Обчислювальні ресурси системи повинні забезпечувати швидку обробку великих обсягів даних у реальному часі. Важливим аспектом є енергоефективність системи, особливо якщо вона призначена для тривалого використання без доступу до постійного джерела живлення.

Система має бути стійкою до збоїв та мати можливість автоматичного відновлення після аварій. Важливо передбачити механізми резервного копіювання даних, щоб уникнути їх втрати в разі непередбачуваних ситуацій. Крім того, система повинна мати високий рівень доступності та мінімізувати час простою.

Для зручності користувачів, система повинна бути простою в установці та налаштуванні. Документація та технічна підтримка повинні бути доступними і зрозумілими, щоб користувачі могли без проблем використовувати всі можливості системи.

#### Підтримка користувачів

Необхідно забезпечити наявність ефективної технічної підтримки, включаючи гарячу лінію, онлайн-чат та інші канали комунікації. Користувачі

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

повинні мати доступ до навчальних матеріалів, інструкцій та рекомендацій щодо оптимального використання системи.

Таким чином, комп'ютеризована система моніторингу біопоказників для тренування стресостійкості повинна відповідати високим вимогам до функціональності, безпеки, надійності та зручності використання, забезпечуючи при цьому точність та оперативність обробки даних для ефективного управління стресовими станами.

### 1.3 Аналіз можливих рішень поставленого завдання

У процесі розробки комп'ютеризованої системи моніторингу біопоказників для тренування стресостійкості необхідно ретельно аналізувати різні аспекти функціонування системи, технології збору даних, їх обробки та інтерпретації. Важливим кроком є визначення ключових біопоказників, які дозволять ефективно оцінювати рівень стресу користувача. Серед них виділяються електропровідність шкіри, пульс та рівень кисню в крові. Кожен з цих параметрів вимагає особливого підходу до вимірювання, а також відповідної обробки та інтерпретації.

Електропровідності шкіри (ГСР - гальванічної шкіри реакції). Цей показник характеризує активність потових залоз і є чутливим до змін у симпатичній нервовій системі, що часто пов'язано зі стресовими реакціями. Зміна електропровідності шкіри може бути швидко і точно зафіксована за допомогою спеціальних датчиків, які кріпляться до пальців або долонь. Ці датчики вимірюють електричний опір шкіри, який знижується при підвищеній потовиділенні. Для обробки отриманих даних використовується алгоритми фільтрації та нормалізації сигналів, що дозволяє виключити шум і отримати точні показники стресового стану користувача. На рис. 1.2 зображено приклад пристрою для вимірювання електропровідності шкіри.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

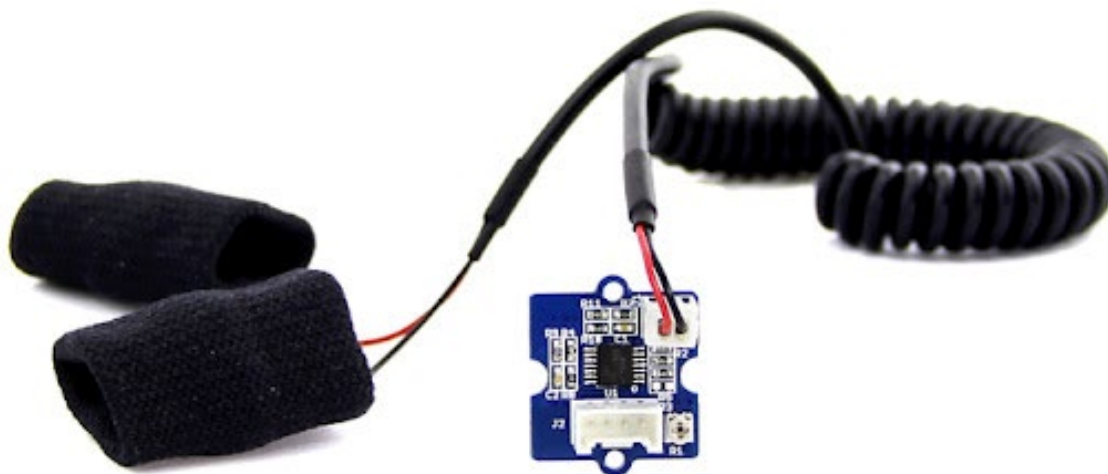


Рисунок 1.2 – Приклад пристрою для вимірювання електропровідності шкіри

Пульс є ще одним важливим показником, який дозволяє оцінювати стан стресу. Частота серцевих скорочень змінюється під впливом стресу, що відображається у вигляді підвищеної частоти пульсу. Знімати дані пульсу можна за допомогою пульсометра, що закріплюється на зап'ясті або пальці. Сучасні пульсометри використовують фотоплетизмографію (PPG) для визначення пульсової хвилі, що дозволяє точно виміряти частоту серцевих скорочень. Аналіз даних пульсу включає обчислення середньої частоти серцевих скорочень, варіабельності серцевого ритму (HRV), яка є індикатором стресового рівня, та інших показників. Для цього застосовуються методи спектрального аналізу та статистичні методи.

Рівень кисню в крові (сатурація) є важливим показником, який відображає стан дихальної системи та загальний фізіологічний стан організму. Під впливом стресу дихання може стати поверхневим або частішим, що впливає на рівень



кисню в крові. Для вимірювання сатурації використовується пульсоксиметрія, яка базується на методі спектрофотометрії. Датчик пульсоксиметра зазвичай кріпиться на пальці і вимірює поглинання світла різних довжин хвиль, що дозволяє визначити рівень насичення крові киснем. Обробка даних сатурації включає фільтрацію сигналу, компенсацію рухових артефактів та інтерпретацію отриманих показників. На рис. 1.3 зображено приклад пристрою для вимірювання рівня кисню та пульсу.

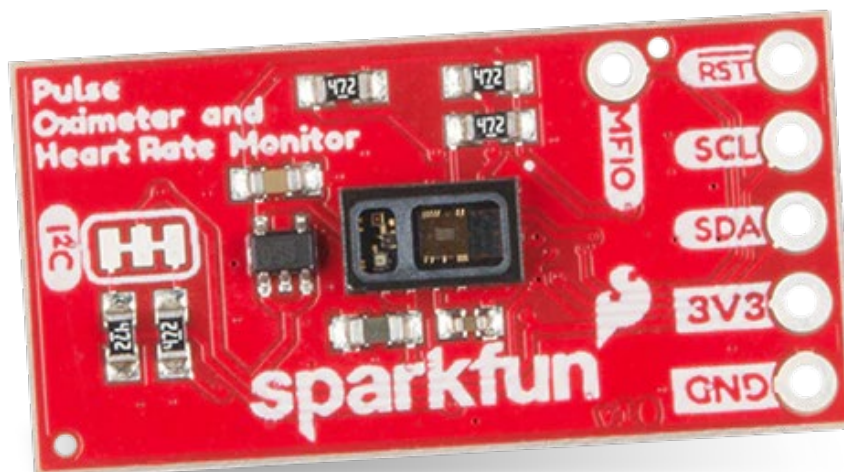


Рисунок 1.3 – Приклад пристрою для вимірювання рівня кисню та пульсу

Інтеграція даних з різних датчиків в єдину систему моніторингу є складним завданням, що вимагає розробки відповідних програмних алгоритмів. Ці алгоритми повинні забезпечувати синхронізацію даних, їх фільтрацію, нормалізацію та аналіз у реальному часі. Важливим аспектом є також розробка зручного інтерфейсу користувача, який дозволить швидко і наочно отримувати інформацію про свій стан стресу та відповідні.

## РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

### 2.1 Розробка узагальненої структури комп'ютеризованої системи моніторингу біопоказників для тренування стресостійкості

Узагальнена структура комп'ютеризованої системи моніторингу біопоказників для тренування стресостійкості надає можливість визначати рівень стресу опираючись на таких показниках як: пульс, рівень кисню та електропровідність шкіри. Вона комбінує різні блоки та пристрої для забезпечення надійного вимірювання та обрахування значень. Узагальнена структурна схема комп'ютеризованої системи зображена на рис. 2.1.



Рисунок 2.1 – Узагальнена структурна схема

Узагальнена структурна схема складається із таких блоків:

- блок вимірювання
- блок моніторингу та обчислення

					<i>КС КРБ 123.127.00.00 ПЗ</i>		
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>			
<i>Розроб.</i>		Рій І.І.			<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевір.</i>		Паляниця Ю.Б.				17	67
<i>Реценз.</i>		Стоянов Ю.М.			<i>РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА</i> <i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. контр.</i>		Тиш Є.В.					
<i>Затверд.</i>		Осухівська Г.М.					

- блок стану системи
- блок виведення в інтернет

Блок вимірювання відповідає за збір інформації про пульс, рівень кисню в крові та електропровідність шкіри. Він складається із датчиків які вимірюють значення та надсилають їх до блоку моніторингу та обчислення. Ці датчики прикріплюються до пальців щоб виміряти значення.

Блок моніторингу та обчислення відповідає за отримання значення із блоку вимірювання та розраховує їх для того щоб кінцеві значення були зрозумілими для користувача. До нього входить мікроконтролер або мікрокомпютер. Він також може включати чи виключати обробку та обчислення даних.

Блок стану системи відповідає за те щоб користувач розумів, чи система на даний момент працює, тобто чи обчислює значення та виводить їх на сайт, чи ні.

Блок виведення в інтернет відповідає за те щоб локальний сайт працював та виводив значення. Також щоб обчислене значення рівня стресу відправлялося на ThinkSpeak графік та виводило його на локальний сервер.

Детальну структурну схему зображено на рис. 2.2.

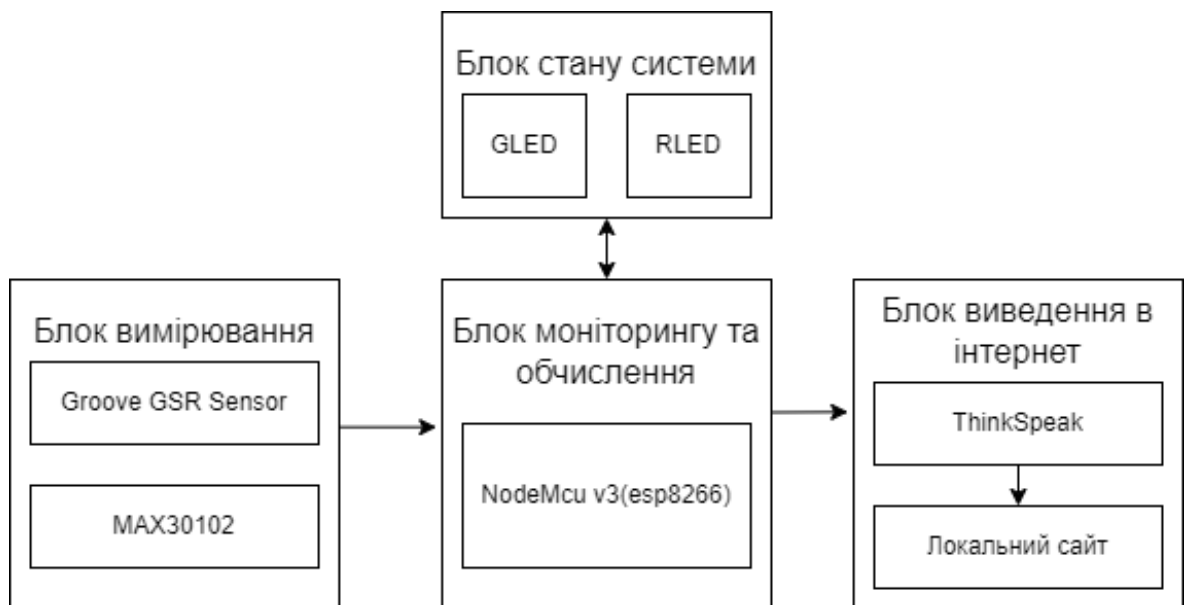


Рисунок 2.2 – Структурна схема

До блоку вимірювання входить два датчика це є датчик електропровідності тіла (Groove GSR sensor) та датчик пульсу і рівню кисню в крові(MAX30102).

До блоку моніторингу та обчислення входить мікроконтролер NodeMcu v3 esp8266.

До блоку сповіщення входять два світлодіода, червоного та зеленого кольору які показують стан системи.

Також є блок виведення в інтернет в ньому будуть локальний сайт і ThinkSpeak.

На рис. 2.3 зображено електрично принципову схему підключень.

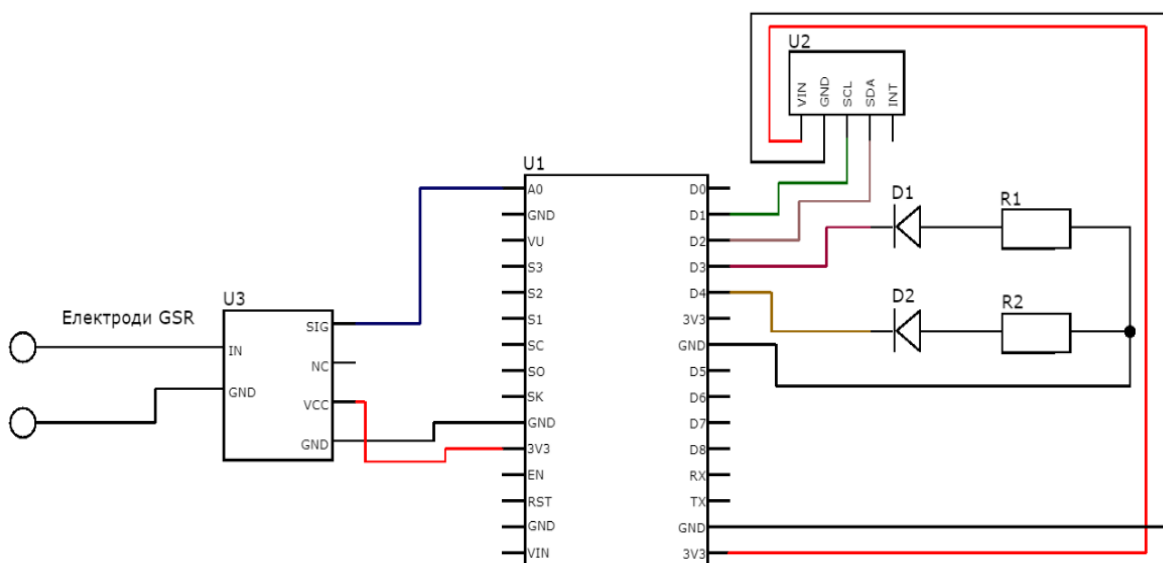


Рисунок 2.3 – Електрично принципова схема

Електрична принципова схема демонструє взаємозв'язок між різними електронними компонентами, серед яких є мікроконтролер ESP8266 NodeMCU, датчик GSR, пульсоксиметр MAX30102 та два світлодіоди із відповідними резисторами.

NodeMCU ESP8266 підключається до джерела живлення через micro USB. Контакти GND мікроконтролера використовуються для заземлення всіх компонентів. Мікроконтролер також має виводи, до яких підключені різні компоненти системи.

Змн.	Арк.	№ докум.	Підпис	Дата

Датчик пульсу MAX30102 підключається до мікроконтролера через кілька контактів. Живлення подається від 3.3V на відповідний вхід датчика, а контакти SCL і SDA датчика підключаються до відповідних контактів мікроконтролера, що забезпечує обмін даними через інтерфейс I2C.

Датчик GSR підключений до мікроконтролера. Живлення подається через контакт VCC до 3.3V, а GND підключений до загального заземлення. Вихідний сигнал від датчика підключений до аналогового входу мікроконтролера, що дозволяє зчитувати значення електропровідності шкіри.

Світлодіоди використовуються для індикації стану системи. Зелений світлодіод підключений до одного з цифрових виходів мікроконтролера через резистор номіналом 220 Ом для обмеження струму, необхідного для роботи світлодіода. Аналогічно, червоний світлодіод підключений до іншого цифрового виходу мікроконтролера через резистор того ж номіналу.

На схемі також зазначені контакти для підключення GSR електродів, що дозволяють знімати показання з поверхні шкіри.

Таким чином, схема забезпечує можливість зчитування біометричних даних та індикацію стану системи за допомогою світлодіодів, об'єднуючи всі компоненти у єдину функціональну систему на базі мікроконтролера ESP8266.

## 2.2 Обґрунтування вибору апаратного забезпечення комп'ютеризованої системи моніторингу біопоказників для тренування стресостійкості

### 2.2.1 NodeMcuV3 esp8266

Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини виконана на основі NodeMcuV3 (ESP8266) [12], який є одним із найпоширеніших мікроконтролерів із вбудованим Wi-fi для проектувань комп'ютерних систем рис. 2.4.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20



Рисунок 2.4 – Зовнішній вигляд NodeMCu

Вибір платформи NodeMCUv3 (ESP8266) для цієї системи обґрунтовується кількома ключовими факторами.

NodeMCUv3 поєднує в собі мікроконтролер ESP8266 та модуль Wi-Fi в одному пристрої. Така інтегрованість сприяє створенню компактних пристроїв, що є важливим для забезпечення мобільності та зручності використання системи моніторингу. Це особливо актуально для портативних рішень, де розміри і вага відіграють важливу роль.

Ще однією перевагою є низька вартість ESP8266, що робить його одним з найбільш доступних мікроконтролерів з вбудованим Wi-Fi. Це значно знижує загальні витрати на розробку та виробництво системи моніторингу, що є

критичним фактором для багатьох проектів, особливо у випадку масового виробництва.

Вбудований модуль Wi-Fi дозволяє забезпечити бездротове підключення до мережі, що є необхідним для передачі даних з біопказників на центральний сервер для подальшого аналізу. Це відкриває можливості для створення розподілених систем моніторингу, де дані збираються з багатьох пристроїв і обробляються централізовано. Налаштовується він в порівнянні із підключеним Wi-Fi модулем до Arduino набагато легше, тому користувачі які будуть задіювати наприклад WebServer обирають Nodemcu.

NodeMCUv3 можна програмувати мовою Lua, яка є простою для вивчення та розуміння, що дозволяє швидко почати розробку програмного забезпечення. Також підтримка Arduino IDE робить цей процес більш доступним для широкого кола розробників, оскільки багато з них вже знайомі з цією платформою.

Наявність вбудованого аналого-цифрового перетворювача (ADC) у ESP8266 значно підвищує його функціональні можливості. Аналого-цифровий перетворювач дозволяє зчитувати аналогові сигнали, які генеруються різноманітними датчиками біопказників, такими як датчики температури, вологості, рівня кисню в крові та інші. Ці датчики зазвичай видають вихідний сигнал у вигляді аналогової напруги, яку потрібно перетворити в цифровий формат для подальшої обробки мікроконтролером.

Вбудований ADC у ESP8266 усуває потребу у використанні зовнішніх перетворювачів для цієї задачі, що спрощує конструкцію пристрою. Це зменшує кількість необхідних компонентів і знижує вартість, оскільки не потрібно купувати окремий ADC і додаткові електронні компоненти для його інтеграції. В результаті, знижується загальна вартість розробки і виробництва системи моніторингу біопказників.

Крім того, ESP8266 має велику спільноту користувачів, що сприяє швидкому вирішенню проблем та обміну досвідом. Велика кількість документації та прикладів коду значно спрощує процес розробки програмного

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

забезпечення, надаючи розробникам доступ до готових рішень і можливість швидко впроваджувати нові функції.

Загальні технічні характеристики мікроконтролера NodeMcuV3 esp8266 наведені у табл. 2.1.

Таблиця 2.1 – загальна технічна характеристика NodeMcuV3 esp8266

Характеристики	Значення
Процесор	ESP8266EX
Архітектура	32-бітний Tensilica Xtensa LX106
Частота процесора	80 МГц (можливе підвищення до 160МГц)
ОЗП(RAM)	64кб інструкційної RAM, 96кб даних RAM
Флеш-пам'ять	4мб
Wi-Fi	2.4 ГГц 802.11 b/g/n
Живлення	5 В через micro USB або 7-12 В через VIN
Робоча напруга	3.3 В на GPIO
Інтерфейси	UART, SPI, I2C
Підтримка протоколів	TCP/IP, HTTP, MQTT
Розмір	Приблизно 58 мм x 31 мм
Роз'єм USB	micro USB
Підтримка програмування	Arduino IDE, NodeMCU (Lua), MicroPython

На платі NodeMCUv3 Lolin зазвичай можуть бути наступні основні виходи живлення:

USB вхід: Платформа може житися через USB-порт, який зазвичай використовується для програмування і живлення. Це зручно для відладки та використання плати в десктопних додатках, де доступний джерело живлення через USB.

Вихід для зовнішнього акумулятора (BAT): Деякі версії NodeMCUv3 можуть мати вихід для зовнішнього акумулятора, який може бути використаний для живлення з батареї. Це дозволяє використовувати плату в мобільних або



вбудованих застосунках, де важливо мати можливість живлення від акумулятора.

Вхідний DC роз'єм: Деякі версії NodeMCUv3 можуть мати роз'єм для вхідного постійного струму (DC), який дозволяє жити платі від зовнішнього джерела живлення, такого як адаптер живлення.

VIN вхід: VIN вхід дозволяє жити платі через зовнішній джерело живлення. Це може бути використано, наприклад, для живлення плати через регульований джерело живлення або інші джерела, що забезпечують необхідні характеристики напруги та струму. Принципова електрична схема представлена на рис. 2.5.

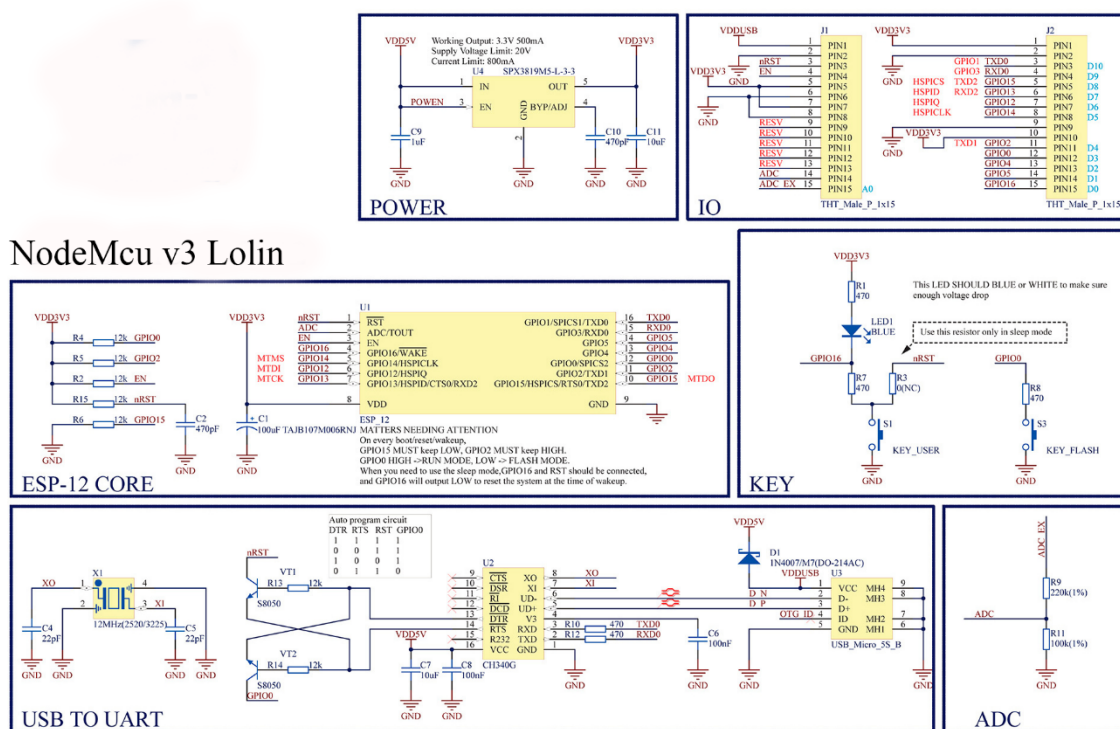


Рисунок 2.5 – Принципова електрична схема плати NodeMCU v3

Основні інтерфейси та роз'єми NodeMcu v3:

- Micro USB порт: Цей роз'єм використовується для живлення мікроконтролера та для програмування його через USB. Ви можете підключити його до комп'ютера або живленого USB-адаптера для живлення.

Змн.	Арк.	№ докум.	Підпис	Дата

- Роз'єм живлення (Vin): Цей роз'єм дозволяє живити NodeMCU зовнішнім джерелом живлення, таким як батарея або джерело живлення постійного струму. Вхідний напруга повинна бути в діапазоні 5-9V.

- Мікрофонний роз'єм (MIC): Цей роз'єм дозволяє підключити мікрофон до мікроконтролера.

- 3.3V та GND (земля): Ці роз'єми надають стабільний струм 3.3V та заземлення відповідно для зовнішніх пристроїв, які потребують живлення або заземлення.

- GPIO (General Purpose Input/Output): NodeMCU має кілька цифрових та аналогових GPIO-пінів, які можна використовувати для підключення різних пристроїв та датчиків.

- UART (Universal Asynchronous Receiver/Transmitter): Цей інтерфейс дозволяє мікроконтролеру взаємодіяти з іншими пристроями за допомогою протоколу UART, такого як зчитування даних з GPS-приймача. GPIO3(RX), GPIO1(TX) – піни на яких працює інтерфейс.

- SPI (Serial Peripheral Interface): Цей інтерфейс дозволяє підключати мікроконтролер до різних периферійних пристроїв, таких як дисплеї або SD-карти, за допомогою протоколу SPI. D5(SCK), D6(MISO), D7(MOSI), D8(SS) – піни на яких працює інтерфейс.

- I2C (Inter-Integrated Circuit): Цей інтерфейс дозволяє підключати мікроконтролер до різних пристроїв за допомогою протоколу I2C, таких як датчики температури та вологості. D1(SCL) та D2(SDA) – піни на яких працює інтерфейс.

- Reset: при натисканні відбувається перезавантаження мікроконтролера. Піни плати NodeMcu v3 представлені на рис. 2.6.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

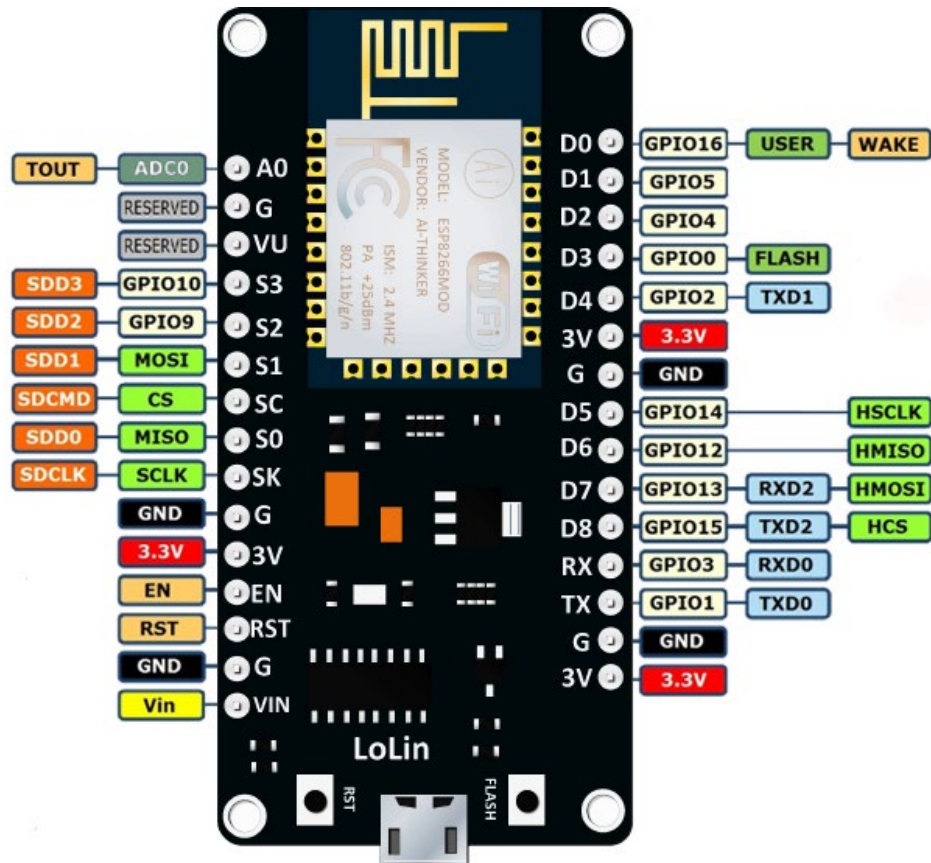


Рисунок 2.6 - Піни плати NodeMcu v3

### 2.2.2 Датчик Groove Gsr

Для вимірювання електропровідності шкіри я обрав датчик Groove GSR v1.2. Датчик Grove GSR Sensor v1.2 є спеціалізованим модулем, розробленим для вимірювання змін електропровідності шкіри, що є індикатором активності потових залоз. Його основне призначення полягає у виявленні емоційних та фізіологічних реакцій людини, таких як стрес, збудження або страх. На рис. 2.3. зображено датчик Groove GSR

Датчик складається з двох основних частин: електродів та електронної схеми. Електроди зроблені з матеріалу, який забезпечує надійний контакт із шкірою і дозволяє точно вимірювати електричний опір. Вони фіксуються на пальцях або інших ділянках шкіри, де їх легко закріпити та де вимірювання будуть найбільш точними. На рис. 2.7 зображено зовнішній вигляд датчика Groove Gsr sensor v1.2.

Змн.	Арк.	№ докум.	Підпис	Дата

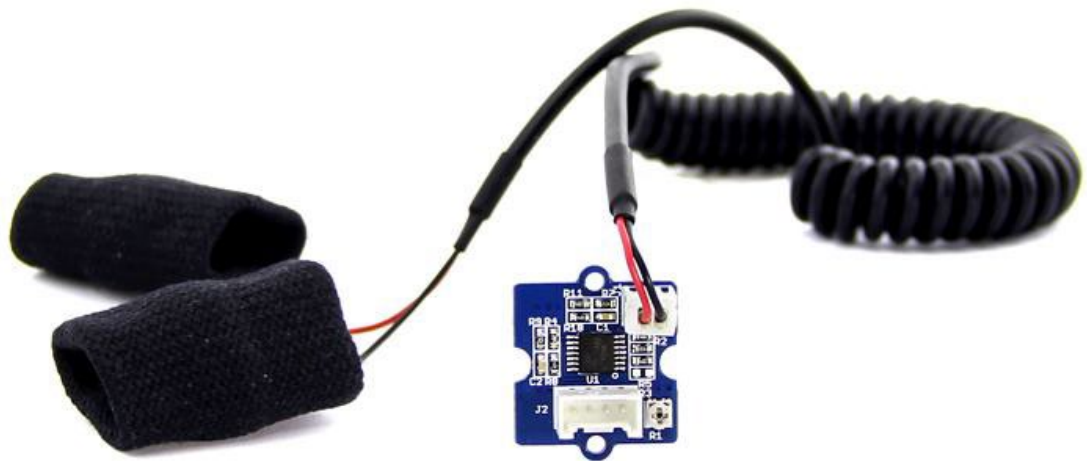


Рисунок 2.7 – Зовнішній вигляд Grove Gsr v1.2

Коли електроди знаходяться в контакті зі шкірою, вони реєструють зміни в електропровідності, викликані активністю потових залоз. Ці залози, під впливом різних емоційних станів чи фізіологічних процесів, виділяють піт, який змінює електропровідність шкіри. Електронна схема датчика перетворює ці зміни в аналоговий сигнал — напругу, яку можна зчитувати за допомогою мікроконтролера, наприклад, Arduino.

Цей аналоговий сигнал відображає рівень електропровідності шкіри і відповідно емоційний чи фізіологічний стан користувача. Мікроконтролер, отримуючи сигнал від датчика, обробляє його за допомогою спеціального програмного забезпечення, яке дозволяє інтерпретувати зміни електропровідності та виводити дані про стан користувача. Це може бути корисно в додатках, які потребують моніторингу стресу, емоційного стану або фізіологічних реакцій.

Датчик інтегрується в екосистему Grove, що робить його сумісним з іншими модулями та контролерами цієї системи. Grove-система розроблена для

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

спрощення процесу підключення та використання різноманітних сенсорів та модулів. Завдяки стандартному конектору Grove, датчик легко підключається до базових плат, таких як Seeduino або Arduino, без необхідності пайки чи складного монтажу.

Крім основних компонентів, датчик має вбудовані калібрувальні резистори, що дозволяють налаштовувати чутливість вимірювань. Це корисно для адаптації датчика до різних умов використання та типів шкіри, забезпечуючи більш точні вимірювання. На рис. 2.8 зображено електрично принципову схему датчика Groove GSR.

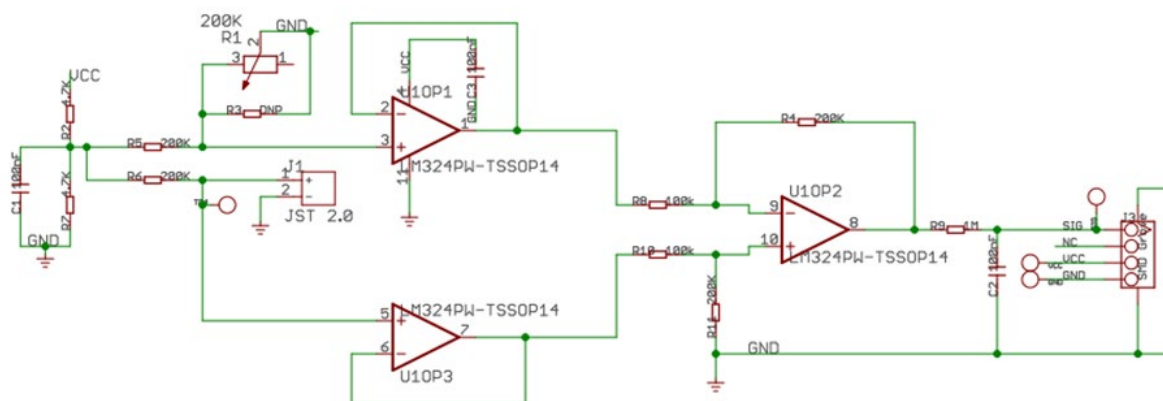


Рисунок 2.8 – електрично принципова схема датчика Groove GSR

Датчик Groove GRS v1.2 має як свої переваги, так і недоліки. Однією з його ключових переваг є простота використання, що робить його зручним для як новачків, так і досвідчених користувачів. Завдяки стандартному інтерфейсу Grove, цей датчик легко інтегрується в різні проекти, що дозволяє швидко отримувати необхідні дані без складної налаштування. Крім того, він характеризується високою точністю та стабільністю вимірювань, що є важливим аспектом для наукових і технічних завдань.

Проте, датчик має і свої недоліки. Одним з них є обмежена функціональність, яка може не задовольнити потреби більш складних або специфічних проектів. Деякі користувачі також відзначають недостатньо

детальну документацію, що може ускладнити процес налаштування та використання датчика. Також варто враховувати його ціну, яка може бути відносно високою у порівнянні з аналогічними продуктами на ринку, що може стати перешкодою для деяких потенційних користувачів.

### 2.2.3 Датчик MAX30102

Датчик MAX30102 [13] є ключовим компонентом комп'ютерної системи моніторингу біоіндикаторів, призначеної для підвищення стресостійкості. Цей оптичний датчик дозволяє вимірювати такі важливі параметри, як рівень кисню в крові ( $SpO_2$ ) і частота серцевих скорочень (ЧСС), які є основними біологічними показниками фізіологічного стану людини. На рис. 2.9 зображено датчик MAX30102.

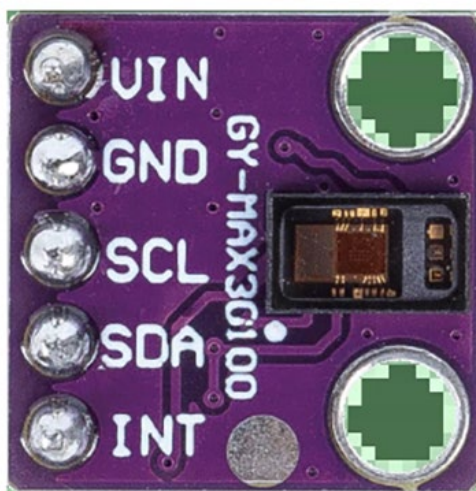


Рисунок 2.9 – Датчик MAX30102

В основі дослідження MAX30102 лежить фотоплетизмографія-метод вимірювання зміни об'єму крові в тканинах за допомогою світла. Датчик включає в себе 2 світлодіода, червоний і інфрачервоний, і фотодетектор. Світлодіоди випромінюють світло в тканині, і, залежно від кількості крові, частина світла поглинається, а частина відбивається і вловлюється фотодетектором. Зміни об'єму крові, викликані скороченнями серця, дозволяють змінювати інтенсивність відбитого світла і визначати пульс. Різниця в поглинанні світла при

використанні червоних і інфрачервоних світлодіодів дозволяє оцінити ступінь насичення крові киснем.

Датчик МАХ30102 відрізняється високою точністю і чутливістю, що дозволяє отримувати достовірні дані при різних умовах освітлення і розташуванні датчика. Його конструкція забезпечує мінімальне енергоспоживання, що робить його ідеальним для використання в системах і портативних пристроях, що вимагають тривалої роботи без підзарядки.

МАХ30102 оснащений інтерфейсом І2С для легкої інтеграції з мікроконтролерами та іншими обчислювальними пристроями. Він також включає в себе функції обробки сигналів, такі як фільтрація перешкод і автоматична корекція зміщення, що значно підвищує якість одержуваних даних і спрощує процес обробки.

Для моніторингу стресостійкості важливо не тільки вимірювати біологічні показники, а й аналізувати їх в динаміці і визначати реакцію організму на стресові ситуації. МАХ30102 володіє високою частотою вимірювань, що дозволяє відстежувати зміни в режимі реального часу. Це особливо важливо при аналізі варіабельності серцевого ритму (ВСР), важливого показника стресу.

Завдяки компактним розмірам і простоті інтеграції МАХ30102 може бути інтегрований в різні носії пристрої, такі як фітнес-трекери, Розумні годинник і медичні Монітори. Це відкриває широкий спектр можливостей для використання в повсякденному житті, забезпечує постійний моніторинг біологічних показників і своєчасне виявлення відхилень від стандартів.

В цілому, використання датчика МАХ30102 в системі моніторингу біоіндикаторів для стресостійких тренувань забезпечує точність, надійність і зручність збору даних. Це важливий аспект для ефективного моніторингу фізіологічного стану людини і своєчасного реагування на стресові фактори.

#### 2.2.4 Arduino IDE

Щоб створити комп'ютерну систему для моніторингу біологічних показників для тренувань стресостійкості, вибір програмного забезпечення є

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

важливим фактором, що визначає ефективність і функціональність кінцевого продукту. Вибір Arduino іде як основного середовища розробки базується на кількох вагомих причинах, які гарантують оптимальне поєднання функціональності, зручності та надійності.

Однією з ключових переваг Arduino іде є відкрите джерело та безкоштовна ліцензія. Це дозволяє використовувати всі функції вашого середовища розробки без додаткових фінансових витрат у жовтні, що є важливим фактором для вашого студентського проекту. Доступність вихідного коду також значно розширює функціональність системи, надаючи можливість змінювати та адаптувати програмне забезпечення до конкретних потреб проекту.

Arduino IDE підтримує широкий спектр мікроконтролерів Arduino, що характеризуються високою надійністю, низьким енергоспоживанням і достатньою обчислювальною потужністю для реалізації завдань, пов'язаних з моніторингом біологічних показників. Arduino IDE забезпечує легку інтеграцію з такими датчиками завдяки великій кількості бібліотек, які спрощують роботу з різними датчиками і модулями. Це дозволяє зосередитись на розробці алгоритмів обробки даних та системної логіки, не витрачаючи занадто багато часу на налаштування обладнання. На рис. 2.10 зображено інтерфейс Arduino IDE.

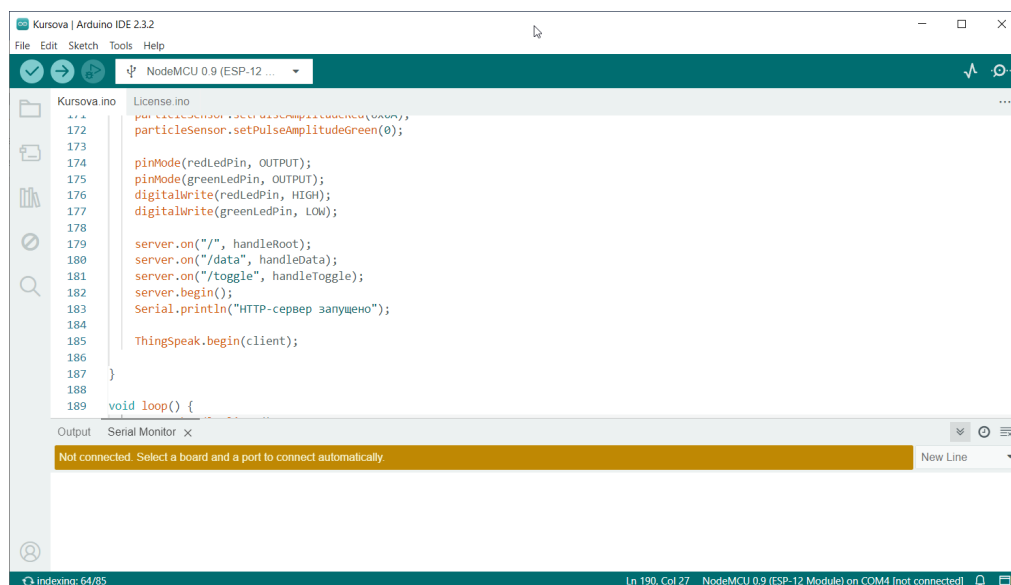


Рисунок 2.10 – Інтерфейс програми Arduino IDE

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31



Простота використання та простота навчання також є вагомим аргументом для вибору Arduino IDE. Інтерфейс середовища розробки інтуїтивно зрозумілий навіть новачкові, що допоможе вам швидко освоїти основи програмування і роботи мікроконтролерів. Інтегроване середовище розробки Arduino надає доступ до великої кількості документації, навчальних матеріалів та прикладів коду для швидкого вирішення нових проблем та проблем. Це особливо важливо в контексті дипломних проєктів, що вимагають незалежних досліджень нових технологій та їх застосування.

Ще однією важливою перевагою є можливість кроссплатформенного використання Ardu декомунізованого середовища розробки Arduino. Середовище розробки підтримує Windows, macOS та Linux, тому ви можете працювати з будь-якою операційною системою без Жовтневої адаптації або використання додаткового програмного забезпечення. Це дозволяє учасникам проєкту використовувати різні операційні системи, що забезпечує гнучкість у виборі робочої платформи і спрощує командну роботу.

Система моніторингу біоіндикаторів для тренувань по стресостійкості дозволяє збирати, обробляти і візуалізувати дані в режимі реального часу. Arduino IDE надає вам усі інструменти, необхідні для вдосконалення цих функцій. Завдяки потужній функції управління послідовним інтерфейсом (UART, I2 деки, SPI), ви можете легко організувати передачу даних між датчиком і комп'ютером або іншими пристроями. Жовтні жовтня. Крім того, можливе розширення функціоналу за рахунок додаткових модулів і плат розширення (щитів), які забезпечують масштабованість проєкту і можливість його подальшого розвитку.

Беручи до уваги всі ці фактори, вибір Arduino IDE в якості основного засобу розробки комп'ютерної системи для моніторингу біологічних показників- це найбільш відкритий вихідний код, широка підтримка мікроконтролерів, велика кількість доступних бібліотек і навчальних матеріалів, зручний інтерфейс. , клієнтський інтерфейс та багато іншого.

## РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

### 3.1 Реалізація проектних рішень

При розробці системи моніторингу біопоказників для тренування стресостійкості слід розробити алгоритм роботи. Для того щоб зрозуміти принцип роботи системи розроблено алгоритм у вигляді блок схеми рис. 3.1.

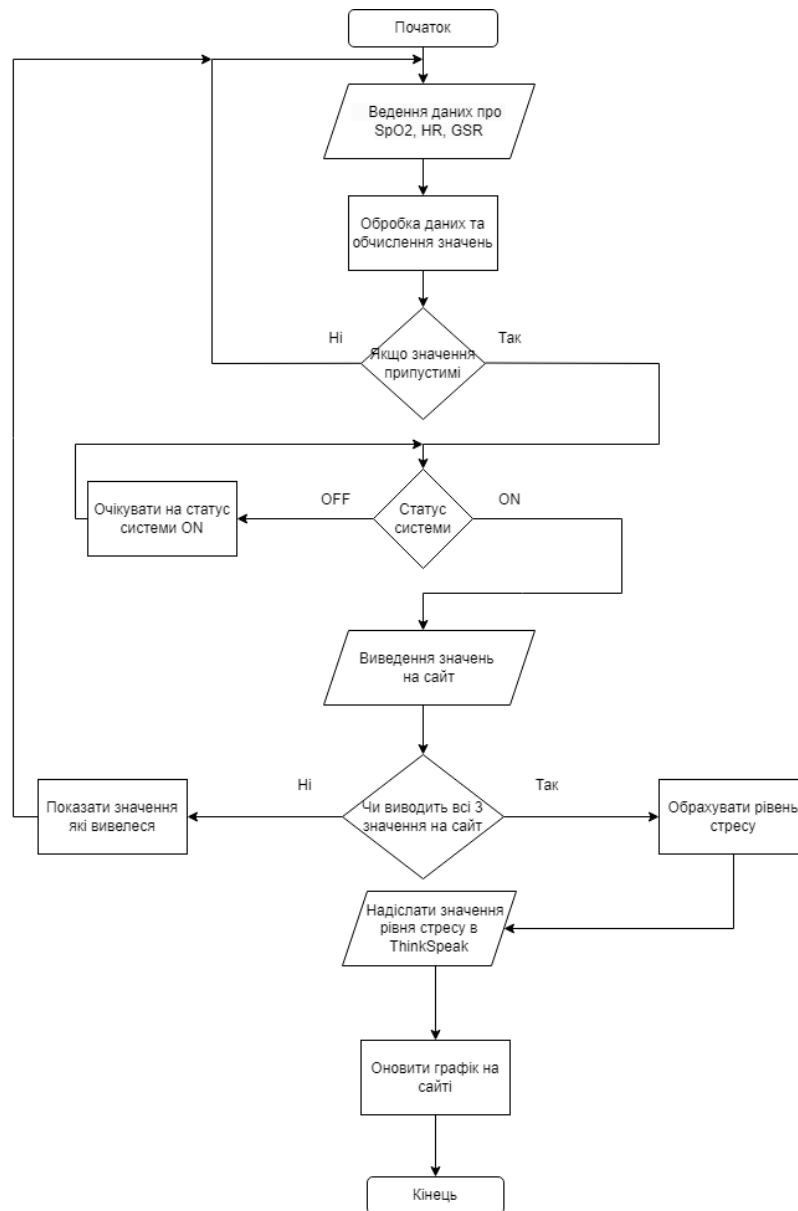


Рисунок 3.1 – Блок схема алгоритму роботи

<i>КС КРБ 123.127.00.00 ПЗ</i>				
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>
<i>Розроб.</i>		Рій І.І.		
<i>Перевір.</i>		Паляниця Ю.Б		
<i>Реценз.</i>		Стоянов Ю.М.		
<i>Н. контр.</i>		Тиш Є.В.		
<i>Затверд.</i>		Осухівська Г.М.		
<i>РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА</i>				
		<i>Літ.</i>	<i>Арк.</i>	<i>Акрушіє</i>
			33	67
<i>ТНТУ, каф. КС, гр. СІ-41</i>				

Використовуючи зазначений алгоритм, система визначає значення рівня кисню в крові, пульс і електропровідність шкіри. Потім він надсилає це значення на сайт і одночасно обчислює рівень стресу та надсилає його на ThinkSpeak.

### 3.1.1 Опис функцій сайту

Для зручності користування комп'ютеризованою системою моніторингу біопоказників для тренування стресостійкості створено сайт який показує в зручному інтерфейсі користувачу дані зчитуваних значень. На рис. 3.2 зображено блок схему роботи сайту.



Рисунок 3.2 – Блок схеми роботи сайту

Змн.	Арк.	№ докум.	Підпис	Дата

### 3.2 Створення формули для обчислення рівня стресу

Для визначення рівня стресу потрібно створити формулу яка би брала значення пульсу, кисню, електропровідності шкіри і обчислювала їх. Також треба вказати у формулі коефіцієнт який буде вказувати важливість того чи іншого значення. Слід також зазначити щоб у формулі визначалося мінімальний та максимальний рівень значень.

Пульс людини у стану спокою дорівнює 70, для максимально у джерелі [14] є формула яка визначає максимальний пульс (3.1).

$$HR_{max} = 220 - (age) \quad (3.1)$$

Тобто для 20 річної людини по формулі буде максимальний пульс 200. Далі слід визначити максимальне та мінімальне значення рівня кисню у крові. У джерелі[15] пише:

— 95%-100%: Нормальний рівень кисню в пульсоксиметрі зазвичай коливається в межах 95%-100%. Примітка: Нормальний рівень може відрізнитися, якщо у вас є захворювання легенів. Ваш лікар може сказати, який рівень є нормальним для вас.

— Нижче 95%: Люди з рівнем SpO2 нижче 95%, наприклад 92%, повинні звернутися до лікаря, оскільки це може бути ознакою гіпоксемії.

— Нижче 80-85%: Мозок страждає, коли рівень SpO2 падає нижче 80-85%.

— Нижче 67%: ціаноз розвивається, коли рівень SpO2 падає нижче 67%.

Візьмемо значення для здорової людини тоді виходить що, значення рівня кисню має бути від 95 до 100 відостків.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

Тепер треба дізнатися який максимальне та мінімальне значення електропровідності шкіри. У джерелі [16] написано, що мінімальні та максимальні значення будуть від 80 до 170.

Після того як відомі мінімальні та максимальні значення можна написати формулу визначення індексу стресу (3.2).

$$\text{Stress index} = 0,5 * \left( \frac{HR-70}{200-70} \right) + 0,2 * \left( 1 - \frac{SpO2}{100} \right) + 0,3 * \left( \frac{GSR-20}{170-20} \right) \quad (3.2)$$

HR– це значення із датчика про пульс.

SpO2 – це значення із датчика про рівень кисню.

GSR – це значення із датчика про електропровідність шкіри.

0,5, 0,2, 0,3 – вагові коефіцієнти, які визначають внесок кожного з параметрів у загальний рівень стресу.

### 3.3 Налаштування Arduino IDE для NodeMcu

Для того щоб програмувати на мікроконтролер NodeMcu у середовищі Arduino IDE, необхідно встановити додатково менеджер плати. Щоб це зробити потрібно зайти в пункті меню “File” і відкрити вікно “Preferences”, у полі “Additional Board Manager URLs” треба написати в цей рядок посилання на файл із розширенням .json рис. 3.3.

[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

[https://raw.githubusercontent.com/esp8266/arduino-esp32/gh-pages/package\\_esp32\\_index.json](https://raw.githubusercontent.com/esp8266/arduino-esp32/gh-pages/package_esp32_index.json)

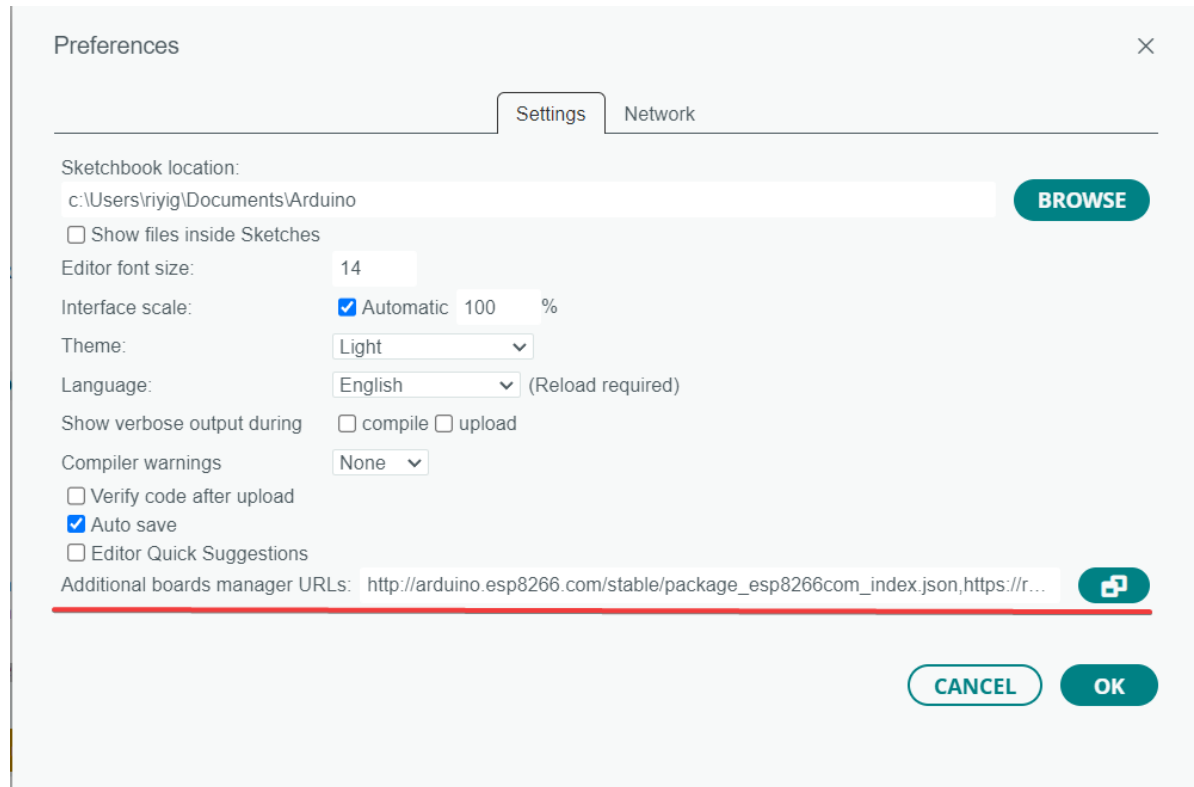


Рисунок 3.3 – Налаштування Arduino IDE

У вікні “Tools” – “Board” там є вікно “Board Manager” у рядку пошуку слід вписати “esp8266” рис 3.4.

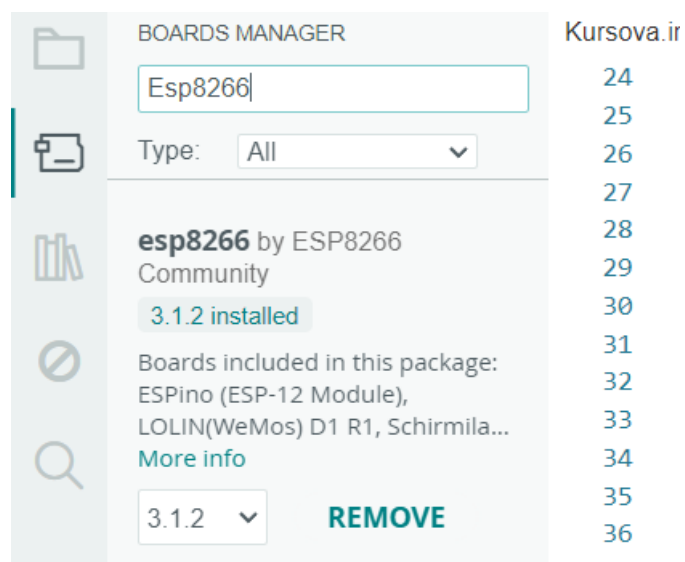


Рисунок 3.4- Процес встановлення розширення плати esp8266

Щоб програмувати на NodeMcu треба зайти знову в меню “Board”  
появиться нова вкладка, в ній слід вказати «NodeMCU 1.0» рис. 3.5.

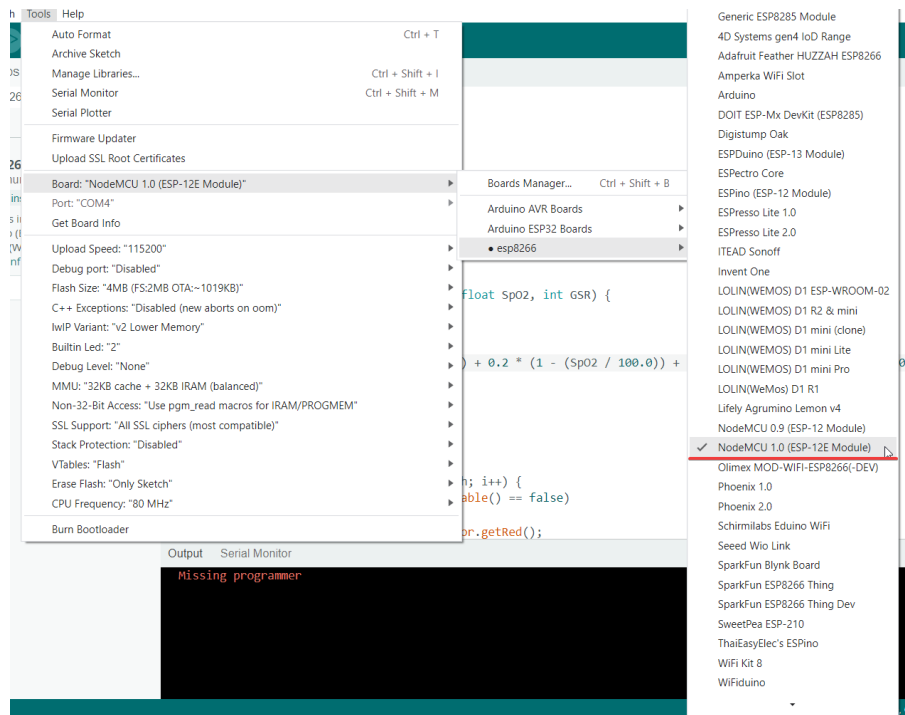


Рисунок 3.5 – Вибір платформи для NodeMcu

### 3.4 Реалізація програмного забезпечення

Для початку слід підключити бібліотеки для користування датчиками MAX30102 та GSR датчика. Також ще слід підключити бібліотеку для того щоб мікроконтролер хостив сайт та відправляв значення на ThinkSpeak. У рис. 3.6 зображено лістинг підключених бібліотек.

```
#include <Wire.h>
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include "MAX30105.h"
#include "spo2_algorithm.h"
#include "heartRate.h"
#include "ThingSpeak.h"
```

Рисунок 3.6 – Лістинг підключених бібліотек

- Wire.h: Ця бібліотека забезпечує роботу з протоколом I2C, який використовується для зв'язку з периферійними пристроями, такими як сенсори.
- ESP8266WiFi.h: Бібліотека для керування Wi-Fi на ESP8266, дозволяє підключати пристрій до бездротової мережі.
- ESP8266WebServer.h: Бібліотека для створення веб-сервера на ESP8266, який може обробляти HTTP-запити.
- MAX30105.h, spo2\_algorithm.h, heartRate.h: Бібліотеки для роботи з сенсором MAX30105, який вимірює пульс (heart rate) та рівень кисню в крові (SpO2).
- ThingSpeak.h: Бібліотека для відправки даних на платформу ThingSpeak, яка дозволяє зберігати та аналізувати дані IoT. У рис. 3.7 зображено лістинг констант та змінних.

```

const char* ssid = "qaz";
const char* password = "12345678";
const char* myWriteAPIKey = "N679BPLFNT9ME30G";
unsigned long myChannelNumber = 2569755;

WiFiClient client;
MAX30105 particleSensor;
ESP8266WebServer server(80);
#define MAX_BRIGHTNESS 255
const int redLedPin = D3;
const int greenLedPin = D4;

bool deviceStatus = false; // false = OFF, true = ON

int GSR_value = 0;
int heartRate = 0;
float SpO2 = 0;
float StressIndex = 0;

uint32_t irBuffer[100];
uint32_t redBuffer[100];
int32_t bufferLength = 100;
int32_t spo2;
int8_t validSP02;
int32_t heartRateData;
int8_t validHeartRate;

```

Рисунок 3.7 – Лістинг змінних та констант



У цьому фрагменті коду оголошуються та ініціалізуються змінні та об'єкти для підключення до Wi-Fi, контролю частоти серцевих скорочень та датчиків кисню в крові (MAX30102) та створення веб-сервера на базі ESP8266.

Спочатку оголошуються 3 змінні типу "const char \*" для SSID, пароля Wi-Fi і API-ключа. Потім для номера каналу визначається змінна "unsigned long".

Для відповідних функцій створюються об'єкти "WiFiClient", "MAX30105" і "Esp8266Webserver". Визначається максимальна яскравість світлодіодного індикатора і оголошуються 2 pin-коду червоного і зеленого світлодіодів і змінна стану пристрою "bool".

Зміни типу "Int", "float" і "uint32\_t" використовуються для збереження значень датчиків, таких як GSR, частота серцевих скорочень, рівень кисню в крові та інші параметри. У рис. 3.8 зображено лістинг функції розрахунку індексу рівня стресу.

```
float calculateStressIndex(int HR, float SpO2, int GSR) {  
    if (GSR > 170) {  
        return 0;  
    }  
    return 0.5 * ((HR - 70) / 200.0) + 0.2 * (1 - (SpO2 / 100.0)) + 0.3 * ((GSR - 80) / 170.0) * 100;  
}
```

Рисунок 3.8 – Лістинг розрахунку індексу рівня стресу

Ця функція обчислює індекс стресу на основі частоти серцевих скорочень (HR), рівня кисню в крові (SpO2) та електропровідності шкіри (GSR). Якщо GSR перевищує 170, індекс стресу дорівнює 0. У рис. 3.9 зображено лістинг функції відправки даних на ThinkSpeak.

```
void sendDataToThingSpeak() {  
    if (StressIndex > 0.1) {  
        ThingSpeak.setField(1, StressIndex);  
        ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);  
    }  
}
```

Рисунок 3.9 – Лістинг функції відправки даних на ThinkSpeak

Ця функція відправляє індекс стресу на платформу ThingSpeak, якщо індекс стресу перевищує 0.1. У рис. 3.10 зображено лістинг функції зчитування даних з датчика.

```
void readSensors() {
    GSR_value = analogRead(A0);

    bufferLength = 100;
    for (int i = 0; i < bufferLength; i++) {
        while (particleSensor.available() == false)
            particleSensor.check();
        redBuffer[i] = particleSensor.getRed();
        irBuffer[i] = particleSensor.getIR();
        particleSensor.nextSample();
    }

    maxim_heart_rate_and_oxygen_saturation(irBuffer, bufferLength, redBuffer, &spo2, &validSPO2, &heartRateData, &validHeartRate);

    if (validSPO2) {
        SpO2 = spo2;
    } else {
        SpO2 = 0;
    }

    if (validHeartRate && heartRateData > 40 && heartRateData < 180) { // Додаткові перевірки на правдоподібність
        heartRate = heartRateData;
    } else {
        heartRate = 0;
    }

    StressIndex = calculateStressIndex(heartRate, SpO2, GSR_value);
}
```

Рисунок 3.10 — Лістинг функції зчитування даних з датчика

В цьому фрагменті коду функція readSensors виконує зчитування даних з різних сенсорів та обробляє їх. Спочатку вона зчитує значення з аналогового сенсора, підключеного до порту A0, і зберігає це значення у змінну GSR\_value.

Далі встановлюється довжина буферу, яка дорівнює 100, і розпочинається цикл, який виконується 100 разів. В кожній ітерації циклу перевіряється, чи доступні нові дані від сенсора (particle sensor), і якщо ні, то викликається функція check для перевірки наявності нових даних. Якщо дані доступні, то зчитуються значення червоного (redBuffer) та інфрачервоного (irBuffer) світла і зберігаються у буфері. Після цього сенсор переходить до наступного зчитування даних.

Після зчитування даних з буферів викликається функція maxim\_heart\_rate\_and\_oxygen\_saturation, яка обчислює рівень кисню у крові (SpO2) та частоту серцевих скорочень (heart rate) на основі зчитаних даних. Результати обчислень зберігаються у змінні spo2, validSPO2, heartRateData та validHeartRate.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		41

Далі відбувається перевірка точності обчислених значень. Якщо рівень кисню у крові є дійсним (validSPO2), то значення SpO2 оновлюється значенням spo2, інакше SpO2 встановлюється у 0. Аналогічно, якщо частота серцевих скорочень є дійсною (validHeartRate) і знаходиться у межах від 70 до 200 ударів на хвилину, то значення heartRate оновлюється значенням heartRateData, інакше встановлюється у 0.

Функція обчислює індекс стресу (StressIndex) на основі частоти серцевих скорочень, рівня кисню у крові та значення з аналогового сенсора, використовуючи функцію calculateStressIndex. На рис. 3.11 зображено лістинг функції яка генерує html сторінку.

```
void handleRoot() {
  String html = "<!DOCTYPE html><html lang='uk'><head><meta charset='UTF-8'><meta name='viewport' content='width=device-width, initial-scale=1.0'>";
  html += "<title>Визначення рівня стресу</title>";
  html += "<style>";
  html += "body {font-family: Arial, sans-serif; background: url('https://iili.io/2mFuvvee.png') no-repeat center center fixed; background-size: cover; text-align: center;";
  html += "#toggleButton {"";
  html += "position: absolute; top: 20px; left: 20px; background-color: #fff666; color: white; font-size: 20px; padding: 10px 20px; border: none; cursor: pointer;";
  html += "};";
  html += "h1 {font-size: 2em; margin-top: 50px;};";
  html += "p {font-size: 1.5em;};";
  html += "#linkContainer {"";
  html += "position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%);};";
  html += "</style>";
  html += "<script>";
  html += "function toggleDevice() {"";
  html += "fetch('/toggle').then(response => response.json()).then(data => {"";
  html += "document.getElementById('toggleButton').innerText = data.status ? 'OFF' : 'ON';";
  html += "document.getElementById('toggleButton').style.backgroundColor = data.status ? '#66ff99' : '#fff666';";
  html += "});";
  html += "};";
  html += "function fetchData() {"";
  html += "fetch('/data').then(response => response.json()).then(data => {"";
  html += "document.getElementById('heartRate').innerText = data.heartRate + ' BPM';";
  html += "document.getElementById('SpO2').innerText = data.SpO2 + '%';";
  html += "document.getElementById('GSR').innerText = data.GSR + ' мкСм';";
  html += "document.getElementById('StressIndex').innerText = data.StressIndex;";
  html += "});";
  html += "};";
  html += "setInterval(fetchData, 5000);";
  html += "</script>";
  html += "</head><body onload='fetchData()'>";
  html += "<h1>ВИЗНАЧЕННЯ РІВНЯ СТРЕСУ</h1>";
  html += "<button id='toggleButton' onclick='toggleDevice()'>ON</button>";
}
```

Рисунок 3.11 – Лістингу функції яка генерує html сторінку

Функція handleRoot створює HTML-сторінку для відображення даних про рівень стресу, пульс, рівень кисню в крові та електропровідність шкіри. Вона встановлює заголовок та стиль сторінки, включаючи фонове зображення, стилі для кнопок та тексту. JavaScript вбудовується для динамічного оновлення даних та перемикання стану пристрою. Коли сторінка завантажується, вона автоматично виконує запит до сервера для отримання даних, які оновлюються кожні 5 секунд. Сторінка також містить iframe, що відображає графік рівня

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

стресу в реальному часі. Після створення HTML-сторінки, вона відправляється клієнту. На рис. 3.12 зображено лістинг функцію перевірки статусу пристрою.

```
void handleData() {
    if (deviceStatus) {
        readSensors();
        sendDataToThingSpeak(); // Відправка на ThingSpeak
        String data = "{\"heartRate\":" + String(heartRate) + ",";
        int displayGSR = (GSR_value > 170) ? 0 : GSR_value;
        data += "\"SpO2\":" + String(SpO2) + ",";
        data += "\"GSR\":" + String(displayGSR) + ",";
        data += "\"StressIndex\":" + String(StressIndex) + "}";
        server.send(200, "application/json", data); // Відправка на сайт
    } else {
        server.send(200, "application/json", "{\"heartRate\":0,\"SpO2\":0,\"GSR\":0,\"StressIndex\":0}");
    }
}
```

Рисунок 3.12 – Лістинг функції перевірки статусу пристрою

В функції `handleData` перевіряється статус пристрою. Якщо пристрій увімкнений, виконуються наступні дії: зчитуються дані із сенсорів і відправляються на платформу ThingSpeak. Далі створюється JSON-рядок із значеннями, отриманими від сенсорів: частотою серцевих скорочень, насиченістю крові киснем (SpO2), значенням шкірно-гальванічної реакції (GSR), яке порівнюється з певним порогом і, якщо воно перевищує цей поріг, встановлюється в нуль, а також індекс стресу. Потім цей JSON-рядок відправляється на сервер.

Якщо ж пристрій вимкнений, на сервер відправляється JSON-рядок із нульовими значеннями для всіх параметрів: частоти серцевих скорочень, насиченості крові киснем, шкірно-гальванічної реакції та індексу стресу. На рис. 3.13 зображено лістинг функції переключення статусу пристрою.

```
void handleToggle() {
    deviceStatus = !deviceStatus;
    digitalWrite(redLedPin, deviceStatus ? LOW : HIGH);
    digitalWrite(greenLedPin, deviceStatus ? HIGH : LOW);
    String status = "{\"status\":" + String(deviceStatus) + "}";
    server.send(200, "application/json", status);
}
```

Рисунок 3.13 – Лістинг функції переключення статусу пристрою

У функції `handleToggle` здійснюється перемикання стану пристрою. Спочатку інвертується поточний стан змінної `deviceStatus`. Потім, залежно від нового стану цієї змінної, на пін червоного світлодіода подається або низький, або високий сигнал, а на пін зеленого світлодіода - відповідно високий або низький сигнал. Далі створюється JSON-рядок, який містить поточний стан пристрою. Нарешті, цей JSON-рядок відправляється клієнту з кодом відповіді 200 і зазначенням, що дані передаються у форматі JSON.

### 3.5 Результат тестування

Після розробки схеми (рис 3.14) та запрограмування мікроконтролера було проведено тестування системи моніторингу біопоказників для тренування стресостійкості.

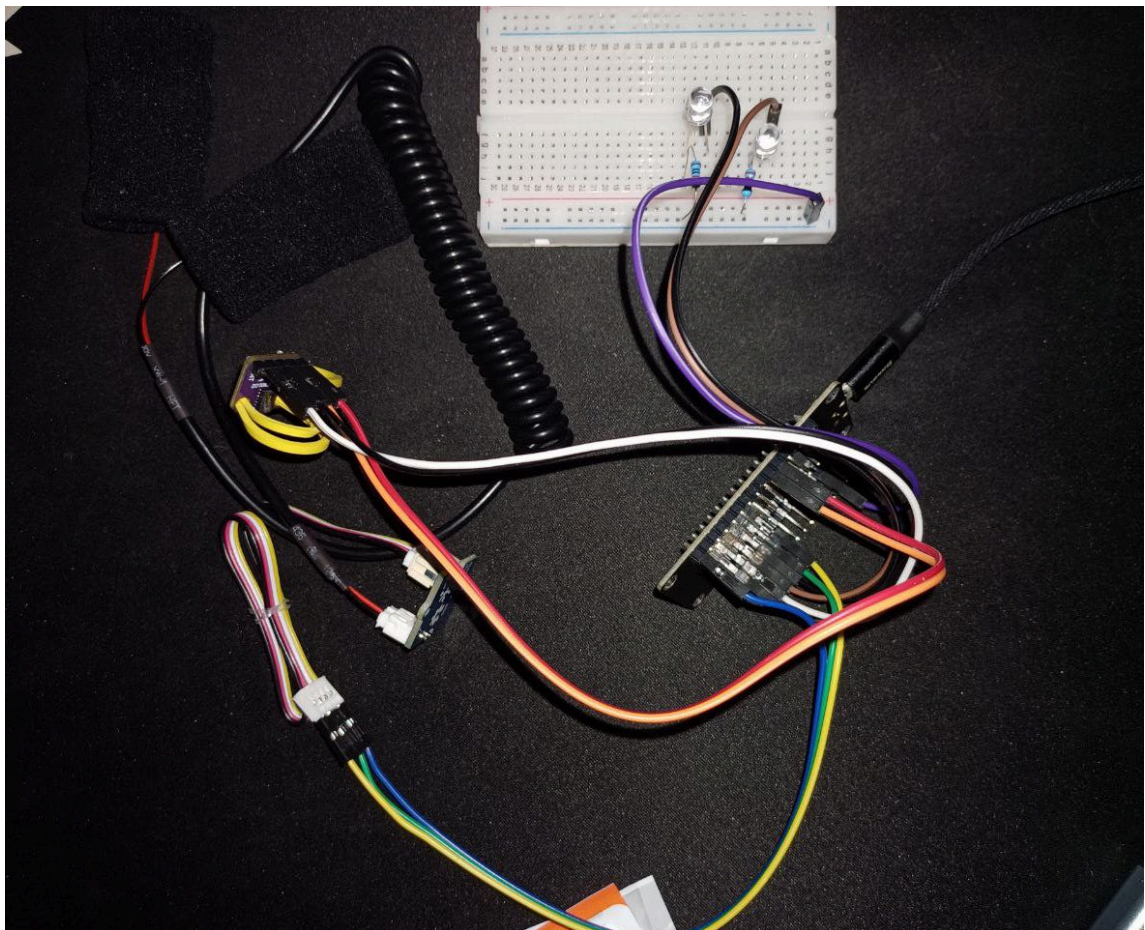


Рисунок 3.14 – Розроблена комп'ютерна системи

Коли мікроконтролер успішно запрограмована, він перезагружається і в серійному моніторі виводить інформацію про успішний запуск системи та виводить ір-адресу через яку можна зайти на сайт (рис. 3.15) якщо ж датчик max30102 не буде підключено виведеться повідомлення про те, що він не підключений і система почне заново перезапускатися.

```

Підключено до WiFi
IP-адреса: 192.168.76.236
HTTP-сервер запущено
    
```

Рисунок 3.15 – Результат про успішний запуск системи

Для того щоб зайти на сайт потрібно бути підключеним до однієї мережі і ввести в пошуковому рядку ір адрес який вивів мікроконтролер в серійному моніторі. Після чого відкриється сайт(рис. 3.16).

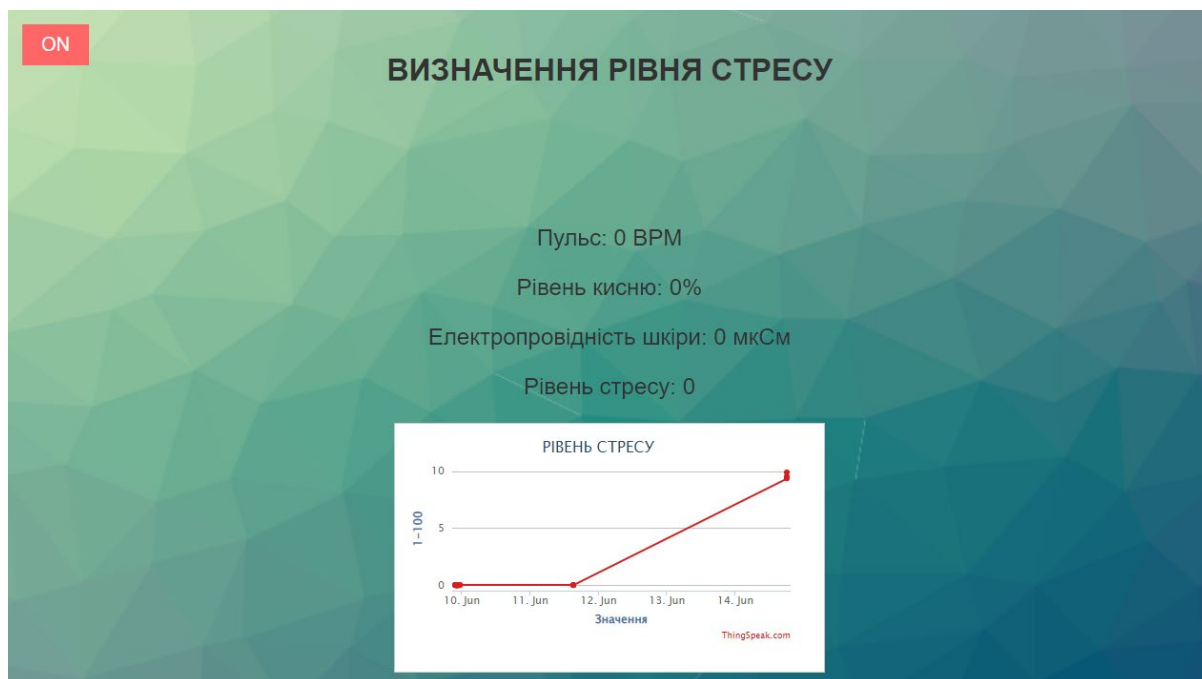


Рисунок 3.16 – Інтерфейс сайту

Щоб на сайті виводилися значення потрібно натиснути кнопку ON після чого загориться зелений світлодіод і дані будуть виводитися у сайті також паралельно будуть виводитися графік ThinkSpeak індексу стресу.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		45

## РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

### 4.1 Фізіологічний вплив факторів існування на життєдіяльність людини

Фізіологічний вплив факторів існування на життєдіяльність людини охоплює численні аспекти, які комплексно визначають її здоров'я, добробут та працездатність. Середовище, в якому людина живе і працює, є вирішальним чинником для багатьох фізіологічних процесів.

Температурний режим середовища значно впливає на терморегуляцію організму, яка є ключовим елементом підтримання гомеостазу. При сприятливих температурних умовах організм ефективно підтримує тепловий баланс, що забезпечує нормальне функціонування всіх систем. В умовах екстремальних температур, організм може зазнавати теплового стресу або переохолодження, що призводить до серйозних порушень в роботі серцево-судинної та нервової систем, зниження фізичної та розумової продуктивності.

Освітлення впливає на біоритми людини та функціонування зорової системи. Адекватний рівень освітлення забезпечує нормальне функціонування зору та знижує ризик розвитку зорової втоми і дискомфорту. Недостатнє або надмірне освітлення може спричинити порушення сну, головний біль, зниження працездатності та негативно позначитися на психоемоційному стані людини.

Якість повітря є критично важливою для дихальної системи. Високий рівень забруднення повітря може призвести до респіраторних захворювань, алергій, зниження імунітету та хронічної втоми. Чисте повітря, навпаки, сприяє кращому насиченню організму киснем, що позитивно впливає на загальний стан здоров'я і працездатність.

Шумове забруднення чинить істотний вплив на нервову систему. Тривалий вплив високого рівня шуму може викликати стрес, втому, дратівливість,

					<i>КС КРБ 123.127.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розроб.</i>		Рій І.І.			<i>РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ ТА ОХОРОНИ ПРАЦІ</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевір.</i>		Паляниця Ю.Б.					46	67
<i>Консульт.</i>		Пилипець М.І.				<i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. контр.</i>		Тиш Є.В.						
<i>Затверд.</i>		Осухівська Г.М.						

зниження концентрації уваги та навіть проблеми зі слухом. Психоемоційний стан значно покращується в умовах акустичного комфорту, що сприяє підвищенню продуктивності та загальному відчуттю добробуту. Взаємодія цих факторів створює комплексне середовище, яке впливає на фізіологічні процеси в організмі. Здорове середовище сприяє підтриманню оптимального функціонування організму, що включає підтримання нормального кровообігу, дихання, обміну речовин та інших ключових функцій. Негативний вплив факторів середовища може мати кумулятивний ефект, поступово підриваючи здоров'я і знижуючи якість життя.

Фізіологічний вплив факторів існування на життєдіяльність людини є багатограним і глибоко пов'язаним з умовами навколишнього середовища. Створення здорового та безпечного середовища є основою для забезпечення добробуту, продуктивності та тривалості життя.

При роботі із комп'ютеризованою системою вимірювання рівня стресу враховано вплив на оператора.

#### 4.2 Заходи щодо запобігання електробезпеки

Заходи електробезпеки в офісі є критично важливими для запобігання нещасним випадкам і забезпечення безпеки працівників. Враховуючи високий ризик ураження електричним струмом, особливо на виробництвах з великим використанням електричного обладнання, заходи електробезпеки повинні бути комплексними та відповідати сучасним стандартам.

По-перше, в офісі розроблено і впроваджено систему організаційних і технічних заходів, спрямованих на попередження нещасних випадків, пов'язаних з електричним струмом. Це включає в себе створення інструкцій з безпечного використання електрообладнання, регулярне проведення навчань і інструктажів для працівників. Ці інструкції мають враховувати специфіку обладнання, яке використовується в конкретному офісі, та потенційні ризики, пов'язані з його експлуатацією.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47



Технічні заходи електробезпеки включають в себе застосування захисного заземлення, занулення, використання захисних відключень та ізоляції струмоведучих частин. Заземлення та занулення дозволяють знизити до мінімуму ризик ураження електричним струмом шляхом створення безпечного шляху для струму у разі пошкодження ізоляції або інших аварійних ситуацій. Використання автоматичних вимикачів та пристроїв захисного відключення (ПЗВ) дозволяє швидко відключати електроживлення у разі виявлення витоку струму або короткого замикання.

Особливо приділяти регулярному технічному обслуговуванню та перевірці стану електрообладнання. Періодичний огляд та тестування дозволяють виявити і усунути потенційні несправності до того, як вони стануть причиною нещасного випадку. Крім того, всі ремонтні роботи на електрообладнанні повинні виконуватися тільки кваліфікованим персоналом, який має відповідну підготовку та допуск до робіт з електроустановками.

В офісі також необхідно забезпечити правильну організацію робочих місць з точки зору електробезпеки. Це включає в себе розміщення електрообладнання таким чином, щоб мінімізувати ризик випадкового контакту з ним, забезпечення достатньої освітленості робочих зон та встановлення попереджувальних знаків та інструкцій у видимих місцях.

Важливість застосування засобів індивідуального захисту (ЗІЗ) для працівників, які працюють з електрообладнанням або в умовах підвищеного електричного ризику. До таких засобів належать діелектричні рукавички, взуття, килимки та інші засоби, що зменшують ризик ураження електричним струмом. Заходи електробезпеки в офісі є багатограним комплексом організаційних та технічних рішень, спрямованих на забезпечення безпечних умов праці та мінімізацію ризиків ураження електричним струмом. Вони включають не лише встановлення відповідного електрообладнання та регулярне проведення технічного обслуговування, але й навчання персоналу правилам поведінки з електроприладами, впровадження сучасних захисних систем, а також дотримання нормативних вимог і стандартів. З метою підвищення ефективності

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

заходів електробезпеки, необхідно періодично проводити інспекції та аудит електричних систем, виявляючи можливі недоліки та ризики.

#### 4.3 Вимоги до профілактичних медичних оглядів для працівників ПК

Профілактичні медичні огляди для працівників, які працюють з персональними комп'ютерами, є невід'ємною частиною охорони праці. Ці огляди спрямовані на збереження здоров'я працівників і запобігання розвитку професійних захворювань, що можуть виникати внаслідок тривалої роботи з комп'ютером. Під час медичних оглядів приділяється зоровому здоров'ю, опорно-руховій системі та стану нервової системи.

Працівники, які проводять багато часу перед комп'ютерним екраном, часто стикаються з різними зоровими проблемами. Під час медичних оглядів лікарі ретельно перевіряють стан очей, оскільки зорове напруження може призвести до розвитку синдрому комп'ютерного зору, сухості очей, порушення гостроти зору та інших офтальмологічних проблем. Лікар-окуліст оцінює гостроту зору, перевіряє наявність рефракційних аномалій, діагностує можливі запальні процеси та надає рекомендації щодо корекції зору та зменшення зорового напруження.

Окрім зорового здоров'я, важливим аспектом є стан опорно-рухового апарату. Тривала робота за комп'ютером часто призводить до неправильного положення тіла, що може спричинити хронічні болі в спині, шиї та кінцівках. Під час медичних оглядів лікарі проводять оцінку стану хребта та суглобів, виявляють наявність остеохондрозу, сколіозу та інших захворювань, пов'язаних із неправильною поставою. На основі отриманих даних лікарі надають рекомендації щодо корекції положення тіла, організації робочого місця, фізичних вправ для зміцнення м'язового корсету та зменшення навантаження на хребет.

Стан нервової системи також є важливим аспектом медичних оглядів для працівників, які працюють з комп'ютерами. Тривала робота з комп'ютером може

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

викликати нервові перенапруження, стресові стани та інші неврологічні проблеми. Під час медичних оглядів лікарі-неврологи оцінюють загальний стан нервової системи, виявляють ознаки перевтоми, стресу та нервового виснаження. Лікарі надають рекомендації щодо методів зменшення стресу, включаючи регулярні перерви, спеціальні вправи для релаксації та психоемоційного розвантаження.

Загальний фізичний стан працівника також підлягає ретельній оцінці під час профілактичних медичних оглядів. Лікарі проводять лабораторні дослідження крові та інших біологічних зразків, оцінюють рівень артеріального тиску, стан серцево-судинної системи та інші життєво важливі показники. Це дозволяє виявити загальні проблеми зі здоров'ям, які можуть впливати на працездатність і загальне самопочуття працівника. Результати медичних оглядів використовуються для розробки індивідуальних рекомендацій щодо режиму праці, організації робочого місця, а також для планування подальших профілактичних заходів.

Регулярність проведення профілактичних медичних оглядів визначається чинним законодавством і може варіюватися залежно від специфіки роботи та стану здоров'я працівника. Головною метою цих оглядів є забезпечення безпеки та здоров'я працівників, запобігання розвитку професійних захворювань і підтримка високого рівня працездатності.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВИСНОВКИ

Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини відображає досягнення поставлених цілей і підтверджує ефективність розробленої системи. У ході дослідження було створено та протестовано програмно-апаратний комплекс, який дозволяє в реальному часі відслідковувати фізіологічні показники, такі як частота серцевих скорочень, рівень стресу та електропровідність тіла. Впровадження системи продемонструвало свою актуальність і значущість для підвищення стресостійкості як у професійному спорті, так і в повсякденному житті.

Розроблена система дозволяє моніторити стан користувача. Аналіз отриманих даних показав, що використання системи сприяє зниженню рівня стресу і покращенню загального фізичного стану користувачів, що підвищує їхню здатність адаптуватися до стресових ситуацій.

Отримані результати підтверджують доцільність впровадження подібних систем у різні сфери діяльності, де необхідна висока стресостійкість. Система може бути корисною для професійних спортсменів, військовослужбовців, співробітників служб екстреної допомоги та інших професій, які пов'язані з високими рівнями стресу. Завдяки інтеграції сучасних технологій моніторингу та аналізу біопоказників, розроблена система відкриває нові можливості для підвищення якості тренувань і покращення здоров'я користувачів.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. E – Learning TNTU. URL: <https://dl.tntu.edu.ua/> (дата звернення 24.04.24).
2. Осухівська Г.М., Тиш Є.В., Луцик Н.С., Паламар А.М. Методичні вказівки до виконання кваліфікаційних робіт здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль, ТНТУ. 2022. 28 с.
3. Palamar A., Palamar M., Osukhivska H. Real-time Health Monitoring Computer System Based on Internet of Medical Things. CEUR Workshop Proceedings, 3rd International Workshop on Information Technologies: Theoretical and Applied Problems (ITAP 2023), Ternopil, Ukraine, Opole, Poland, November 22–24, 2023. Vol. 3628. P. 106-115.
4. Паламар А.М., Сомін Д.С. Комп'ютеризована система моніторингу рівня насичення киснем крові людини на основі ІоМТ. Матеріали ХІ науково-технічної конференції «Інформаційні моделі, системи та технології» Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 13-14 грудня 2023 року), Тернопіль: ТНТУ, 2023. С. 177.
5. Луцик Н. С. , Литвиненко Я. В. , Лупенко С. А. , Зозуля А. М . Програмний комплекс для морфологічного аналізу та аналізу серцевого ритму з підвищеною інформативністю URL: <https://itce.vntu.edu.ua/index.php/itce/article/download/266/353>.(дата звернення 27.04.24).
6. Palaniza Y.B., Shadrina H.M., Khvostivskiy M.O., Dediv L.Ye., Dozorska O.F . Main theoretical basis of biosignals modeling URL: [https://elartu.tntu.edu.ua/bitstream/lib/24343/1/13\\_1-2\\_39-44\\_48.pdf](https://elartu.tntu.edu.ua/bitstream/lib/24343/1/13_1-2_39-44_48.pdf) (дата звернення: 28.04.24).
7. Lupenko Serhii, Lutsyk Nadiia, Yasniy Oleh, Sobaszek Łukasz. Statistical analysis of human heart rhythm with increased informativeness. URL: <https://sciendo.com/article/10.2478/ama-2018-0047?content-tab=abstract>. (дата звернення: 28.04.24).
8. Heart rate. URL: [https://en.wikipedia.org/wiki/Heart\\_rate](https://en.wikipedia.org/wiki/Heart_rate) (дата звернення 29.04.24).

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

9. Сатурація. URL: <https://uk.wikipedia.org/wiki/Сатурація> (дата звернення 01.05.24).
10. Lipovski, Jack. G. Introduction to Microcontrollers: Architecture, Programming, and Interfacing for the Freescale 68HC12 (Academic Press Series in Engineering). Vancouver, 2014, с.376.
11. NodeMCU . URL: <https://en.wikipedia.org/wiki/NodeMCU> (дата звернення 09.05.24).
12. ESP8266 NodeMCU WiFi Module User Manual. URL: <https://handsontec.com/dataspecs/module/esp8266-V13.pdf> (дата звернення(14.05.24).
13. MAX30102 High-Sensitivity Pulse Oximeter and Heart-Rate Sensor for Wearable Health URL: <https://www.analog.com/media/en/technical-documentation/datasheets/max30102.pdf> (дата звернення 27.05.24).
14. Grove - GSR Sensor. URL: [https://files.seeedstudio.com/wiki/Grove-GSR\\_Sensor/res/Grove-GSR\\_Sensor\\_WiKi.pdf](https://files.seeedstudio.com/wiki/Grove-GSR_Sensor/res/Grove-GSR_Sensor_WiKi.pdf) (дата звернення 08.06.24).
15. What Are Blood Oxygen Levels? URL: [https://www.medicinenet.com/what\\_are\\_blood\\_oxygen\\_levels/article.htm](https://www.medicinenet.com/what_are_blood_oxygen_levels/article.htm) (дата звернення 10.06.24).
16. Galvanic Skin Response. URL: <https://www.sciencedirect.com/topics/computer-science/galvanic-skin-response> ( дата звернення 13.06.24).
17. Владика, В., Величко, Д., Осухівська, Г. М. Технології передавання даних в системі «Цифрова лікарня». Матеріали VII науково-технічної конференції „Інформаційні моделі, системи та технології “, Тернопіль: ТНТУ, 2019. С. 109.
18. Микитишин А. Г., Митник М. М., Стухляк П. Д., Пасічник В. В. Комп'ютерні мережі. Книга 1 [навчальний посібник]. Львів : «Магнолія 2006», 2013. 256 с.
19. Хвостівська, Л. В., Осухівська, Г. М., Хвостівський, М. О., Шадріна, Г. М. Розвиток методів та алгоритмів обчислення періоду стохастичних біомедичних сигналів для медичних комп'ютерно-діагностичних систем. Вісник Національного технічного університету України Київський політехнічний інститут. Серія: Радіотехніка. Радіоапаратобудування, №79, 2019. С. 78-84.

					<i>КС КРБ 123.127.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

Додаток А  
Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана  
Пулюя Факультет комп'ютерно-інформаційних систем і програмної  
інженерії

Кафедра комп'ютерних систем та мереж

**«Затверджую»**

Завідувач кафедри КС

Осухівська Г.М.

“ 09 ” 02 \_\_\_\_\_ 2024р.

Комп'ютеризована система моніторингу біопоказників для тренування  
стресостійкості людини

ТЕХНІЧНЕ ЗАВДАННЯ

на 7 листках

**Вид робіт:**

Кваліфікаційна робота

**На здобуття освітнього ступеня**

**«Бакалавр» Спеціальність 123**

**«Комп'ютерна інженерія»**

**«УЗГОДЖЕНО»**

Керівник кваліфікаційної роботи

\_\_\_\_\_ к.т.н. ст. викл. Паляниця Ю.Б.

“ 08 ” 02 \_\_\_\_\_ 2024р

**«ВИКОНАВЕЦЬ»**

Студент групи СІ-41

\_\_\_\_\_ Рій І.І.

“ 07 ” 02 \_\_\_\_\_ 2024р

Тернопіль 2024



# 1 Загальні відомості

## 1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості людини».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.127.00.00

## 1.2 Виконавець

Студент групи СІ-41, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерної інженерії, Тернопільського національного технічного університету імені Івана Пулюя, Рій Ігор Ігорович.

## 1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№4/7-408).

## 1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання роботи – 24.02.2024р.

Плановий термін завершення виконання роботи – 26.06.2024р.

## 1.5 Порядок оформлення та представлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу

здійснюється у відповідності до чинних норм та правил ІСО, ЄСКД, ЄСПД та ДСТУ.

Пред'явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи.

Попередній захист кваліфікаційної роботи відбувається при готовності роботи на 90%, наявності пояснювальної записки та графічного матеріалу.

Пред'явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

## 2 Призначенні і цілі створення розробки

### 2.1 Призначення розробки

Комп'ютеризована система моніторингу біопоказників для тренування стресостійкості є дуже простою у користуванні та інтуїтивно зрозумілою.

До складу системи повинні входити як апаратна складова, так і програмна.

### 2.2 Мета створення системи

Комп'ютеризована система призначена для тренування стресостійкості людини, щоб покращити стресостійкість людини в критичних ситуаціях.

### 2.3 Характеристика об'єкту

Система проєктується для інформування користувача, що включає в себе:

- Розробку структурної схеми;

- Розробку електричної принципової схеми;
- Розробку блок-схеми алгоритму роботи;
- Розробку блок-схеми роботи сайту;

### 3 Вимоги до системи

#### 3.1 Загальні вимоги

Комп'ютеризована система автоматичного пермикання постійного живлення на резервне з веденням обліку відключень повинна забезпечити:

- Компактність: Система не повинна займати багато простору;
- Низька собівартість;
- Можливість вираховувати значення;

##### 3.1.1 Вимоги до структури та функціонування системи

Структура системи моніторингу біопоказників для тренування стресостійкості:

- NodeMcu – Основний компонент, який виконує обчисленням, зчитуванням обробкою та виводом даних.
- MAX30102 – Датчик який зчитує дані пульсу та рівень кисню в крові.
- Groove GSR - Датчик який зчитує дані про електропровідність шкіри.

##### 3.1.2 Вимоги в функціонуванні системи

Система функціонує у режимі, який налаштував користувач.

##### 3.1.3 Вимоги до надійності системи

Система повинна бути захищена від фізичних та механічних пошкоджень як на апаратному, так і на програмному рівнях. Її надійність має гарантувати відновлення функціонування у випадку збоїв апаратних компонентів або програмного забезпечення. Імовірність безперебійної роботи системи повинна бути не менше 95%.

#### 3.1.4 Вимоги до функцій та задач, які виконує система

Основна функціональна задача системи полягає у зчитуванні значень, пульсу, рівня кисню в крові та електропровідності шкіри. Також із цих значень обраховується рівень стресу.

#### 3.1.5 Вимоги до апаратного забезпечення

- Режими роботи і умови експлуатації вибраних елементів повинні відповідати вказаним в ТЗ
- Елементна база має забезпечувати необхідні режими роботи системи

### 4 Вимоги до документації

Документація повинна відповідати вимогам

ЄСКД та ДСТУ. Комплект документації складається з:

- Пояснювальної записки
- Графічного матеріалу:
  - Структурної схеми програми;
  - Електрично-принципової схеми;
  - Блок-схеми алгоритму роботи;

- Блок-схеми роботи сайта;

\*Примітка: у комплект документації можуть вноситися зміни та доповнення в процесі розробки

## 5 Техніко-економічні показники

Планована собівартість проєкту повинна становити не більше 1,5 тисяч гривень.

\*Примітка: собівартість може змінюватися під час розрахунку в процесі розробки.

## 6 Стадії та етапи проєктування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи

№ етапу	Назва виконання кваліфікаційної роботи	Термін виконання
1	Розробка та затвердження технічного завдання	01.02.24-04.02.24
2	Аналіз можливих рішень поставленого завдання	05.02.24-11.02.24
3	Розробка структурної, функціональної та електричної принципової схеми	06.05.24-11.05.24
4	Вибір компонентів та програмного забезпечення для комп'ютеризованої системи	12.05.24-15.05.24
5	Проектування апаратного забезпечення системи моніторингу біопоказників	16.05.24-20.05.24

6	Реалізація програмного забезпечення комп'ютеризованої системи	<i>21.05.24-25.05.24</i>
7	Тестування системи моніторингу біопоказників	<i>26.05.24-30.05.24</i>
8	Оформлення кваліфікаційної роботи	<i>01.06.24–13.06.24</i>
9	Попередній захист кваліфікаційної роботи	<i>14.06.24</i>
10	Захист кваліфікаційної роботи	<i>26.06.24</i>

#### 7 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у технічне завдання можуть вноситися зміни та доповнення.

Додаток Б  
Перелік елементів





## Додаток В

### Лістинг

```
#include <Wire.h>
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include "MAX30105.h"
#include "spo2_algorithm.h"
#include "heartRate.h"
#include "ThingSpeak.h"
const char* ssid = "qaz";
const char* password = "12345678";
const char* myWriteAPIKey = "N679BPLFNT9ME30G";
unsigned long myChannelNumber = 2569755;
WiFiClient client;
MAX30105 particleSensor;
ESP8266WebServer server(80);
#define MAX_BRIGHTNESS 255
const int redLedPin = D3;
const int greenLedPin = D4;
bool deviceStatus = false; // false = OFF, true = ON
int GSR_value = 0;
int heartRate = 0;
float SpO2 = 0;
float StressIndex = 0;
uint32_t irBuffer[100];
uint32_t redBuffer[100];
int32_t bufferLength = 100;
int32_t spo2;
int8_t validSPO2;
int32_t heartRateData;
int8_t validHeartRate;
float calculateStressIndex(int HR, float SpO2, int GSR) {
    if (GSR > 170) {
        return 0;
    }
    return 0.5 * ((HR - 70) / 200.0) + 0.2 * (1 - (SpO2 / 100.0)) + 0.3 *
((GSR - 80) / 170.0) * 100;
}
void readSensors() {
    GSR_value = analogRead(A0);

    bufferLength = 100;
    for (int i = 0; i < bufferLength; i++) {
        while (particleSensor.available() == false)
            particleSensor.check();
        redBuffer[i] = particleSensor.getRed();
        irBuffer[i] = particleSensor.getIR();
        particleSensor.nextSample();
    }
}
```

```

    maxim_heart_rate_and_oxygen_saturation(irBuffer, bufferLength,
redBuffer, &spo2, &validSPO2, &heartRateData, &validHeartRate);

    if (validSPO2) {
        SpO2 = spo2;
    } else {
        SpO2 = 0;
    }
    if (validHeartRate && heartRateData > 40 && heartRateData < 180) { //
Додаткові перевірки на правдоподібність
        heartRate = heartRateData;
    } else {
        heartRate = 0;
    }
    StressIndex = calculateStressIndex(heartRate, SpO2, GSR_value);
}
void sendDataToThingSpeak() {
    if (StressIndex > 0.1) {
        ThingSpeak.setField(1, StressIndex);
        ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
    }
}
void handleRoot() {
    String html = "<!DOCTYPE html><html lang='uk'><head><meta charset='UTF-
8'><meta name='viewport' content='width=device-width, initial-scale=1.0'>";
    html += "<title>Визначення рівня стресу</title>";
    html += "<style>";
    html += "body {font-family: Arial, sans-serif; background:
url('https://iili.io/JmFUvee.png') no-repeat center center fixed;
background-size: cover; text-align: center; color: #333;}";
    html += "#toggleButton {";
    html += "position: absolute; top: 20px; left: 20px; background-color:
#ff6666; color: white; font-size: 20px; padding: 10px 20px; border: none;
cursor: pointer;";
    html += "}";
    html += "h1 {font-size: 2em; margin-top: 50px;}";
    html += "p {font-size: 1.5em;}";
    html += "#linkContainer {";
    html += "position: absolute; top: 50%; left: 50%; transform: translate(-
50%, -50%);}";
    html += "</style>";
    html += "<script>";
    html += "function toggleDevice() {";
    html += "fetch('/toggle').then(response => response.json()).then(data =>
{";
    html += "document.getElementById('toggleButton').innerText = data.status
? 'OFF' : 'ON';";
    html += "document.getElementById('toggleButton').style.backgroundColor =
data.status ? '#66ff99' : '#ff6666';";
    html += "});";
}

```

```

    html += "});";
    html += "function fetchData() {";
    html += "fetch('/data').then(response => response.json()).then(data =>
{";
    html += "document.getElementById('heartRate').innerText = data.heartRate
+ ' BPM';";
    html += "document.getElementById('SpO2').innerText = data.SpO2 + '%';";
    html += "document.getElementById('GSR').innerText = data.GSR + '
мкСм';";
    html += "document.getElementById('StressIndex').innerText =
data.StressIndex;";
    html += "});";
    html += "};";
    html += "setInterval(fetchData, 5000);";
    html += "</script>";
    html += "</head><body onload='fetchData()>";
    html += "<h1>ВИЗНАЧЕННЯ РІВНЯ СТРЕСУ</h1>";
    html += "<button id='toggleButton'
onclick='toggleDevice()>ON</button>";
    html += "<div id='linkContainer'>";
    html += "<p>Пулс: <span id='heartRate'></span></p>";
    html += "<p>Рівень кисню: <span id='SpO2'></span></p>";
    html += "<p>Електропровідність шкіри: <span id='GSR'></span></p>";
    html += "<p>Рівень стресу: <span id='StressIndex'></span></p>";
    html += "<iframe width='450' height='260' style='border: 1px solid
#cccccc;'
src='https://thingspeak.com/channels/2569755/charts/1?bgcolor=%23ffffff&colo
r=%23d62020&dynamic=true&max=100&min=0&results=100&title=%D0%A0%D0%86%D0%92%
D0%95%D0%9D%D0%AC+%D0%A1%D0%A2%D0%A0%D0%95%D0%A1%D0%A3&type=line&xaxis=%D0%9
7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%BD%D1%8F&yaxis=1-100'></iframe>";
    html += "</div></body></html>";
    server.send(200, "text/html", html);
}

void handleData() {
    if (deviceStatus) {
        readSensors();
        sendDataToThingSpeak(); // Відправка на ThingSpeak
        String data = "{\"heartRate\":\"" + String(heartRate) + ",";
        int displayGSR = (GSR_value > 170) ? 0 : GSR_value;
        data += "\"SpO2\":\"" + String(SpO2) + ",";
        data += "\"GSR\":\"" + String(displayGSR) + ",";
        data += "\"StressIndex\":\"" + String(StressIndex) + "\"";
        server.send(200, "application/json", data); // Відправка на сайт
    } else {
        server.send(200, "application/json",
"{\"heartRate\":0,\"SpO2\":0,\"GSR\":0,\"StressIndex\":0}");
    }
}

void handleToggle() {
    deviceStatus = !deviceStatus;
}

```

```

digitalWrite(redLedPin, deviceStatus ? LOW : HIGH);
digitalWrite(greenLedPin, deviceStatus ? HIGH : LOW);
String status = "{\"status\":\"" + String(deviceStatus) + "\"}";
server.send(200, "application/json", status);
}
void setup() {
  Serial.begin(115200);

  WiFi.begin(ssid, password);
  Serial.println("Підключення до WiFi...");
  int retries = 0;
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
    retries++;
    if (retries == 20) {
      Serial.println("Не вдалося підключитися до WiFi. Будь ласка,
перевірте свої облікові дані.");
      return;
    }
  }
  Serial.println("\nПідключено до WiFi");

  Serial.print("IP-адреса: ");
  Serial.println(WiFi.localIP());
  pinMode(A0, INPUT);
  if (!particleSensor.begin(Wire, I2C_SPEED_FAST)) {
    Serial.println("MAX30102 не знайдено. Будь ласка, перевірте
проводку/живлення.");
    while (1);
  }
  particleSensor.setup();
  particleSensor.setPulseAmplitudeRed(0x0A);
  particleSensor.setPulseAmplitudeGreen(0);
  pinMode(redLedPin, OUTPUT);
  pinMode(greenLedPin, OUTPUT);
  digitalWrite(redLedPin, HIGH);
  digitalWrite(greenLedPin, LOW);
  server.on("/", handleRoot);
  server.on("/data", handleData);
  server.on("/toggle", handleToggle);
  server.begin();
  Serial.println("HTTP-сервер запущено");
  ThingSpeak.begin(client);
}
void loop() {
  server.handleClient();
}

```