Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

_________Інститут електроенергетики_________
(інститут)
_________факультет інформаційних технологій__________
(факультет)
Кафедра_____ інформаційних технологій та комп’ютерної інженерії __________
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня ___ магістра__
(бакалавра, спеціаліста, магістра)

Студента _______________________ Білана Вячеслава Романовича ___________________
(ПІБ)
академічної групи _____________123м-19-1_____________________________________
(шифр)
спеціальності _____123 «Комп’ютерна інженерія»_______________________________
(код і назва спеціальності)
за освітньо-професійною програмою _____________«Комп’ютерна інженерія»_______
(офіційна назва)
на тему ______«Комп’ютерна система моніторингу прийому медикаментів і відстеження стану пацієнтів»_________________________________
(назва за наказом ректора)

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	проф. Гнатушенко В.В			
розділів:				
Загальні положення про моніторингові та МІС				
Проектування системи				
Розроблення ПЗ				

Рецензент	доц. Коротенко Л.М.			

Нормоконтролер	проф. Цвіркун Л.І.			

Дніпро
2020

ЗАТВЕРДЖЕНО:
завідувач кафедри
___інформаційних технологій___
______та комп’ютерної інженерії___
(повна назва)

____________________ _Гнатушенко В.В._
(підпис) (прізвище, ініціали)

«_____»_____________ 2020_року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня ____магістр ______
(бакалавра, спеціаліста, магістра)

студенту ______Білана В.Р.________________ **академічної групи** ____123м-19-1___
(прізвище та ініціали) (шифр)

спеціальності __________123 «Комп’ютерна інженерія»___________________________

за освітньою-професійною програмою _____«Комп’ютерна інженерія»__________ ___

на тему _«Комп’ютерна система моніторингу прийому медикаментів і відстеження стану пацієнтів»____

затверджену наказом ректора НТУ «Дніпровська політехніка» від_________ №_________

Розділ	**Зміст**	**Термін виконання**
Розділ 1	Комп’ютерна система моніторингу прийому медикаментів і відстеження стану пацієнтів	5.11.2020
Розділ 2	Комп’ютерна система моніторингу прийому медикаментів і відстеження стану пацієнтів	30.11.2020
Розділ 3	Розроблення програмного забезпечення	10.12.2020

Завдання видано ___________________ ______Гнатушенко В.В.________
(підпис керівника) (прізвище, ініціали)

Дата видачі ______10 вересня .2020 р.__________

Дата подання до екзаменаційної комісії ____10.12.2020 р.____

Прийнято до виконання ______________ ________Білан В.Р.__________
(підпис студента) (прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 76 стор., 35 рис., 5 додаток, 21 джерело.

Об'єкт дослідження: Комп'ютерна система моніторингу прийому медикаментів і відстеження стану пацієнтів

Мета кваліфікаційної роботи: Огляд та аналіз існуючих моніторингових систем та додатків для прийому медикаментів.

Кваліфікаційна робота присвячена актуальній задачі вивчення ефективності застосування метрик оцінки якості процесів і сервісів при використанні різних фреймворків для розробки веб-сайтів.

У вступі розглянута актуальність даної роботи, та мета.

У першому розділі описані загальні положення про моніторингові системи, мобільні технології в охороні здоров'я, методи діагностики, статистичні методи та спектральні методи аналізу.

У другому розділу розглянута структура системи моніторингу, призначення та ціль системи, інформаційна модель та її опис та проектування бази даних.

У заключному третьому розділі було розглянуто середовище розробки та програмна платформа, був показаний інтерфейс програми

Практична цінність даної роботи полягає у використанні розробленого програмного забезпечення для аналізу життєвих показників людини.

Список ключових слів: МОНІТОРИНГОВА СИСТЕМА, МЕДИЧНА ІНФОРМАЦІЙНА СИСТЕМА, JAVASRIPT, NODE.JS, WEBSTROM

ABSTRACT

Explanatory note: 76 pages, 35 figures, 5 ext., 21 sources.

Object of research: Computer system for monitoring medication and tracking patients

Purpose of the qualification work: Review and analysis of existing monitoring systems and applications for medication.

Qualification work is devoted to the urgent task of studying the effectiveness of the application of metrics for assessing the quality of processes and services when using different frameworks for website development.

The introduction considers the relevance of this work and its purpose.

The first section describes general provisions on monitoring systems, mobile healthcare technologies, diagnostic methods, statistical methods and spectral analysis methods.

The second section considers the structure of the monitoring system, the purpose and purpose of the system, the information model and its description and database design.

In the final third section, the development environment and software platform were considered, and the program interface was shown

The practical value of this work is to use the developed software for the analysis of human vital signs.

List of keywords: MONITORING SYSTEM, MEDICAL INFORMATION SYSTEM, JAVASRIPT, NODE.JS, WEBSTROM

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ 7
ВСТУП 8
1 ЗАГАЛЬНІ ПОЛОЖЕНННЯ ПРО МОНІТОРИНГОВІ ТА МЕДИЧНО ІНФОРМАЦІЙНІ СИСТЕМИ 10
1.1 Загальні положення про моніторингові системи та медичні інформаційні системи 10
1.2 Огляд і аналіз існуючих моніторингових систем 14
1.3 Smart пристрої та додатки для прийому медикаментів 19
1.4 Мобільні технології в охороні здоров'я 24
1.5 Методи діагностики 25
1.6 Статистичні методи 28
1.7 Геометричні методи (варіаційна пульсометрія) 30
1.8 Автокореляційний аналіз 32
1.9 Спектральні методи аналізу ВСР 34
2. ПРОЕКТУВАННЯ СИСТЕМИ 38
2.1 Призначення та ціль системи 38
2.2 Структура системи моніторгнгу 38
2.3 Об’єкт управління та опис систем моніторингу 40
Синхронізація всіх потоків 42
2.3.1 Характеристика об'єкта автоматизації 42
2.3.2 Вимоги до структури та функціонування системи 42
2.3.3 Питання які вирішує система моніторингу 43
2.4 Інформаційна модель та її опис 43
2.5 Проектування бази даних 48

3. РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 51
3.1 Обґрунтування вибору технологій для розробки 51
3.1.1 Середовище розробки JavaScript 51
3.1.2 Програмна платформа Node.js 54
3.1.3 Середовище розробки WebStorm 56
3.2 Опис програмного інтерфейсу системи 57
3.3 Опис контрольного прикладу реалізації 61
ВИСНОВКИ 67
ПЕРЕЛІК ПОСИЛАНЬ 68
ДОДАТОК А 71
ДОДАТОК Б 72
ДОДАТОК В .... **Ошибка! Закладка не определена.**
ДОДАТОК Г .... **Ошибка! Закладка не определена.**
ДОДАТОК Д .... **Ошибка! Закладка не определена.**

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

МІС – Медична інформаційна система

БД – База даних

HTTP - HyperText Transfer Protocol

CSS – Cascading Style Sheets –каскадні стилі сторінок

CVS - Concurrent Versions System

JS – JavaScript – мова програмування

MVC - Model-View-Controller

ВСТУП

Об'єктом дослідження - є Комп'ютерна система моніторингу прийому медикаментів і відстеження стану пацієнтів. На сьогоднішній день на інформаційному ринку існує безліч автоматизованих медичних інформаційних систем, що володіють різними якісними і кількісними характеристиками. Однак вибір медичної найбільш придатною системи для моніторингу стану пацієнта викликає певні труднощі, оскільки відсутня цілісна картина, що дозволяє судити, яка автоматизована медична інформаційна система найбільш повно охоплює весь процес моніторингу.

Сучасні медичні організації виробляють і накопичують величезні обсяги даних. Від того, наскільки ефективно ця інформація використовується лікарями, керівниками, які керують органами, залежить якість медичної допомоги, загальний рівень життя населення, рівень розвитку країни в цілому і кожного її територіального суб'єкта зокрема. Тому необхідність використання великих, і при цьому ще постійно зростаючих, обсягів інформації при вирішенні діагностичних, терапевтичних, статистичних, управлінських та інших завдань, обумовлює сьогодні створення інформаційних систем в медичних установах.

Для повноцінного моніторингу пацієнта була розроблена система, яка починає відстежувати всю історія пацієнта від прийому, консультацій і призначення курсів лікування до самого лікування.

За допомогою даної дипломної роботи автоматизується моніторинг стану пацієнта, тобто зменшуватися витрати часу на цей процес.

Створювана медична інформаційна система повинна в першу чергу оптимізувати збір інформації, допомагати лікаря при постановці діагнозу, сприяти зменшенню лікарських помилок і усунення їх негативних наслідків, а також виконувати свою основну функцію - моніторинг стану пацієнта.

Метою дипломної роботи є розробка системи моніторингу стану пацієнта і прийому медикаментів, яка буде здійснювати функцію пошуку, відображати основні

дані, параметри по кожному пацієнту, також користувач зможе бачити інформацію про програму і т.д.

1 ЗАГАЛЬНІ ПОЛОЖЕНННЯ ПРО МОНІТОРИНГОВІ ТА МЕДИЧНО ІНФОРМАЦІЙНІ СИСТЕМИ

1.1 Загальні положення про моніторингові системи та медичні інформаційні системи

На даний момент в Україні інформаційні системи в медичних установах дуже поширені. Практично у всіх медичних установах, як державних, так і приватних, моніторинг пацієнтів здійснюється за допомогою інформаційних систем. Ця область постійно розвивається, і формуються нові підходи до даної діяльності. Метою діяльності є надання висококваліфікованої та спеціалізованої медичної допомоги населенню за рахунок забезпечення доступності сучасних методів діагностики, ефективного використання унікального обладнання.

Вважаю, що талонна система попереднього запису та попередній запис на прийом до лікарів-консультантів по телефону є не досить зручною і не відповідає сучасним тенденціям розвитку IT- технологій. В даний час актуальним було б впровадження інформаційної системи в медицину, з метою створення не тільки електронної попереднім записом (електронна реєстратура), яку населення може забезпечити зі своїх персональних комп'ютерів в будь-який час доби, а й «медичного інтернету», який буде містити бази медичних даних, мультимедійні навчальні сервери, віртуальні атласи і підручники, демонстрації клінічних випадків, медичні бібліотеки, електронні версії журналів, описи науково-дослідних проектів, програмне забезпечення для обробки зображень і багато іншого. Так само актуальним було б впровадження медичної інформаційної системи по веденню електронної історії хвороби.

Впровадження інформаційної системи підтримки надання медичної допомоги в діяльності медичних установ призведе до забезпечення висококваліфікованої, сучасними, зручними, а головне швидкої медичної допомоги населенню. Похід до лікарні перестане асоціюватися з постійними і нескінченними чергами, важкими для пенсіонерів, інвалідів, майбутніх матерій і просто для людей, які вміють цінувати час. Всі ці нововведення давно назріли і їх необхідно втілювати в життя. Медицина повинна бути якісною і сучасною.

Головною метою розробників медичних інформаційних систем (МІС) є комплексне вирішення проблем збору та аналізу інформації, а також завдань управління лікувально-профілактичної та фінансовою діяльністю установи.

Відповідно, для кожного підрозділу лікувально-профілактичного закладу (ЛПЗ) необхідно вирішувати конкретні завдання, не забуваючи про інтеграцію прикладних проектів в єдине ціле.

Проте складнощі з визначенням пріоритетного напряму діяльності ЛПЗ істотно ускладнюють розробку і впровадження медичних інформаційних систем.

Наприклад, створювана медична інформаційна система повинна в першу чергу оптимізувати збір інформації, допомагати лікаря при постановці діагнозу, сприяти зменшенню лікарських помилок і усунення їх негативних наслідків.

Однак в реальних умовах на лікувально-профілактичну діяльність істотно впливає ефективність управління установою в цілому. Особливо добре це помітно в українських стаціонарах, де ситуація ускладнюється соціально-економічними факторами.

Таким чином, автоматизація ЛПЗ повинна проходити на всіх рівнях кожного підрозділу. У зв'язку з цим, при впровадженні медичних інформаційних систем необхідно приділяти увагу і діагностичної складової (в тому числі функції підтримки прийняття рішень), і статистичної (аналіз різнорідних даних, складання звітів для страхових компаній), і економічної (оптимізація фінансової діяльності організації).

Автоматизація діяльності ЛПЗ - це перспективний підхід не тільки для організації документообігу, а й при оптимізації всієї діяльності медичної організації.

Однак, незважаючи на очевидні переваги медичних інформаційних систем, а також значний досвід їх успішного застосування в західних і європейських клініках, в Україні автоматизація ЛПЗ відбувається повільніше, ніж можна було б очікувати.

Основною причиною такого стану справ багато експертів вважають недостатню інформованість лікарів і керівників медичних установ про МІС і результати їх використання в медицині.

Другим за значимістю фактором, який негативно впливає на впровадження медичних інформаційних систем в українських лікувально-профілактичних

установах, є слабка технічна база ЛПЗ. Неможливо застосовувати сучасні технології там, де відсутні комп'ютери або персонал не володіє необхідними технічними даними для їх використання.

Слід зазначити, що хоча експерти і кажуть про недостатньо ефективне застосування комп'ютерних технології в українських медичних установах, багато хто з них стверджують, що в даний час ситуація змінюється на краще.

Сучасний рівень розвитку засобів Internet - комунікацій і мобільного зв'язку відкриває нові напрямки підвищення ефективності роботи в різних напрямках медицини. Один з перспективних підходів до вдосконалення медичного обслуговування полягає в застосуванні для спостереження за станом окремих категорій пріоритетних пацієнтів систем цілодобового on-line моніторингу в віддаленому режимі. У цьому випадку пацієнт і спеціалізований медичний сервіс знаходяться на відстані від декількох сотень метрів до тисяч кілометрів. Використовуючи засоби віддаленого моніторингу, медичний заклад цілодобово відстежує стан стану пацієнтів в кожен момент часу з можливістю їх аналізу та реєстрації в своїй базі даних, що дозволяє оперативно приймати рішення про надання медичної допомоги. Медична організація, що володіє технологією віддаленого моніторингу, отримує незаперечні переваги в підвищенні ефективності своєї роботи, розширюючи комплекс сервісних послуг для пацієнтів, які потребують медичного супроводу поза стаціонаром, контролю тяжкохворих пацієнтів відділень реанімації, інтенсивної терапії та приймальних відділень медичних закладів

Завдання системи віддаленого моніторингу полягає в передачі всіх потоків реєстрованих з пацієнта (групи пацієнтів) даних на значну відстань без втрати якості і синхронності в режимі реального часу. Така передача повинна бути реалізована протягом тривалого часу з наданням відповідним службам медустанови інформаційної картини стану пацієнта. У всіх випадках система моніторингу виконує безперервну реєстрацію, віддалену передачу, обробку та відображення медичному персоналу повного набору електрофізіологічних параметрів стану пацієнта в синхронізований масштабі часу. Моніторинг виконується цілодобово в режимі реального часу з підтримкою різних форматів представлення діагностичної

інформації. Всі потоки даних мають адресне призначення і виводяться на термінали профільних служб організації або засоби мобільної комунікації медичного персоналу. Система віддаленого моніторингу виконують комплексування електрофізіологічних характеристик пацієнта для формування узагальненої оцінки його стану. Реєстровані показники пацієнта можуть ретранслювати провідною медорганізації профільним медустанов або декільком службам однієї установи для організації спільних досліджень і прийняття рішень. Таким чином, актуальність завдання віддаленого моніторингу та наявність сучасних розвинених засобів комунікації, їх доступність і поширеність є передумовами до розробки та впровадження в медичну практику нових засобів дистанційної діагностики.

Поєднуючи в собі переваги існуючих систем телемедицини та телеметрії, віддалений моніторинг має переваги, що дозволяють синтезувати «інформаційний портрет» хворого (групи пацієнтів) в режимі реального часу, не прив'язуючись до його фізичного місцезнаходженням. При цьому реалізуються такі унікальні можливості:

Висока швидкість передачі даних. Реальний трафік високошвидкісного широкосмугового Internet досягає 100 Мб / сек, що дозволить забезпечити стійку передачу електрофізіологічних показників стану пацієнтів спеціалізованим службам медичних закаладів в режимі реального часу.

Комплексування показників. У медичний заклад одночасно надходять потоки даних про стан відразу декількох пацієнтів. Реєстровані дані проходять три рівня комплексування. На першому рівні об'єднуються дані з електрофізіологічного устаткування по кожному пацієнту, на другому рівні - дані декількох пацієнтів і на третьому - дані декількох медустанов, де розміщуються пацієнти. Для комплексування потоків медичної інформації використовуються апаратно-програмні засоби ,що пройшли апробацію при вирішенні аналогічних завдань віддаленого зв'язку.

Ведення бази даних моніторингу. Всі потоки даних, що надходять в організацію по віддаленій зв'язку, автоматично зберігаються в медичній базі даних і строго асоціюються з медичними картами спостережуваних пацієнтів. Це дозволяє

фахівцям реалізувати Постпроцессорние аналіз зареєстрованої інформації з можливістю її роздруківки, архівування та експорту на зовнішні носії. Таким чином, по кожному «віддаленого» пацієнтові лікаря завжди буде доступний повний обсяг нативной інформації про динаміку зміни показників хворого за весь час спостереження.

Розвинений сервіс on-line аналізу вхідних потоків інформації. У процесі спостереження віддалених пацієнтів фахівцям надаються можливості коментування (текст, аудіо запис голосу лікаря) реєструється інформації в будь-який момент часу, виконання оперативних роздруківок окремих ділянок записи, реалізації операцій спеціалізованої обробки даних і ряд інших можливостей, що підвищують ефективність on-line аналізу. Отже, візуальне сприйняття динаміки електрофізіологічних показників доповнюється широким функціональним сервісом її обробки.

Безперервне, цілодобове моніторування. Стан пацієнта реєструється у вигляді сукупності електрофізіологічних показників з частотою оцифрування не менше 100 Гц. Сумарний потік інформації при одночасному спостереженні 15 пацієнтів становить до 1 Мб / сек. Всі дані цілодобово надходять на Інтернет - шлюзи медорганізації і розподіляються по відділах, відображаючись на терміналах для оперативного аналізу.

Синхронізація всіх потоків в режимі реального часу. Всі потоки даних, що реєструються з «віддалених» пацієнтів в режимі реального часу, строго синхронізовані. Всі події записуються в пам'ять системи автоматично і коментуються фахівцями в on-line режимі. Такий підхід дозволяє отримати якісну оцінку клінічної ситуації по кожному пацієнту.

1.2 Огляд і аналіз існуючих моніторингових систем

WinPatientExpert (Рис 1.1) - це сучасна медична база даних яка дозволяє асоціювати різнорідні формати даних діагностики з реєстраційною карткою пацієнта. Будучи універсальною базою даних для медичних програм, забезпечує роботу з декількома незалежними базами даних пацієнтів, кожна з яких може зберігати велику кількість карток пацієнтів. Крім стандартних полів картки пацієнта, користувачеві

доступна можливість повної настройки картотеки виходячи з конкретної медичної специфіки. Головною особливістю СУБД WinPatientExpert є можливість асоціації з реєстраційної картки пацієнта результатів його обстеження на різному медичному обладнанні. При цьому дані можуть відрізнятися не тільки за призначенням, але і формою подання - текст, графіка, анімація, звук, відео. Таким чином, вибравши пацієнта в нашій БД, ви отримуєте всю пов'язану з ним інформацію. Комп'ютерна обробка медичних даних з використанням функціональних сервісів WinPatientExpert являє собою гнучкий і зручний інструмент перегляду, автоматизованої обробки, експорту та імпорту медичної інформації різного статусу.

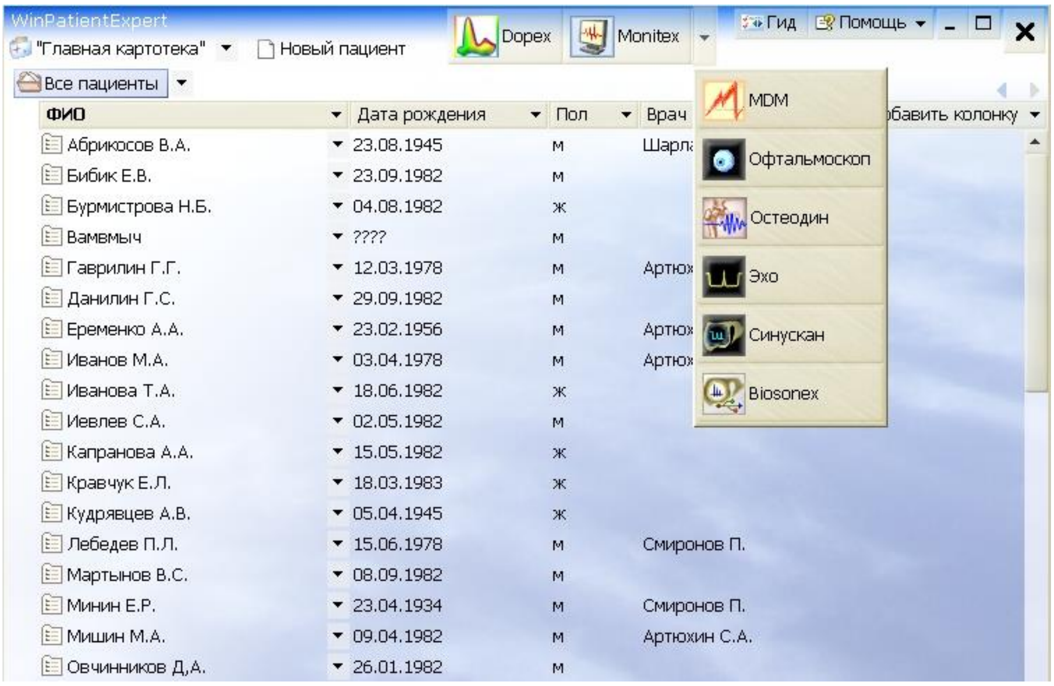

Рисунок 1.1 – Головне меню WinPatientExpert

Можливості програми. Інтегрована система обробки даних WinPatientExpert дозволяє одночасно маніпулювати різнорідними даними, отриманими в ході обстежень пацієнтів на різному діагностичному обладнанні. Можливість виклику з єдиної бази даних декількох методик обстеження створює унікальні умови для спільного використання різних методів обробки та зберігання даних: тексту, графіків, таблиць, зображень, аудіо- та відео- інформації, асоційованої з даними реєстрації пацієнта.

Наочність і зручність організації представленої в WinPatientExpert інформації досягається за рахунок її багаторівневої ієрархічної структуризації, що забезпечує

високу ефективність доступу медичного персоналу до даних з метою їх подальшої деталізації.

Універсальність технології обробки різних форматів медичних даних в WinPatientExpert дозволяє виконувати широкий набір операцій аналізу, редагування, друку експорту та імпорту даних в розподілених обчислювальних системах різної архітектури, включаючи віддалений доступ.

Основні особливості:

- настройка всіх атрибутів призначеного для користувача інтерфейсу (розмір і тип використовуваного шрифту, структура і склад інформації, що відображається, мова інтерфейсу);
- редагування структури реєстраційної карти пацієнта, включаючи склад полів карти, їх найменування, розташування і характеристики;
- завдання параметрів режиму автоматичного збереження даних пацієнта при роботі в умовах нестабільного електроживлення;
- генерація дистрибутива бази даних з встановленими користувачем налаштуваннями;
- можливість скачати нову версію бази даних з сервера компанії;
- автоматична генерації з середовища WinPatientExpert демонстраційних файлів для виставок, навчальних курсів і консультацій;
- копіювання даних на різноманітні зовнішні носії (оптичні, магнітооптичні, магніторезистивні та ін.) з попереднім стисканням інформації;
- реалізація системи вбудованої фільтрації пацієнтів, об'єднаних одним або групою інформаційних критеріїв; автоматизація процедур обстеження за допомогою технології «Гід», що дозволяє в покроковому режимі провести повний цикл обстежень в стислі терміни.

Unimonex (Рис 1.2) - система багатоканального on-line моніторингу пацієнта. Дослідження стану пацієнта проводиться одночасно кількома методиками з розширеними можливостями по обробці, візуалізації і зберігання результатів діагностики. Unimonex дозволяє одночасно реєструвати дані з дев'яти різних приладів, виводячи на екран і зберігаючи в базі даних WinPatientExpert: модуль

доплера, модуль ЕЕГ, модуль ЕМГ, модуль ЕКГ, модуль тонометра, модуль вимірювання інвазивного тиску, модуль капнографи, модуль пульоксіметра, модуль Стабілометр. Крім цього, програма Unimonex забезпечена новим модулем для автоматичного детекції емболії в умовах операційного моніторингу церебральної гемодинаміки.

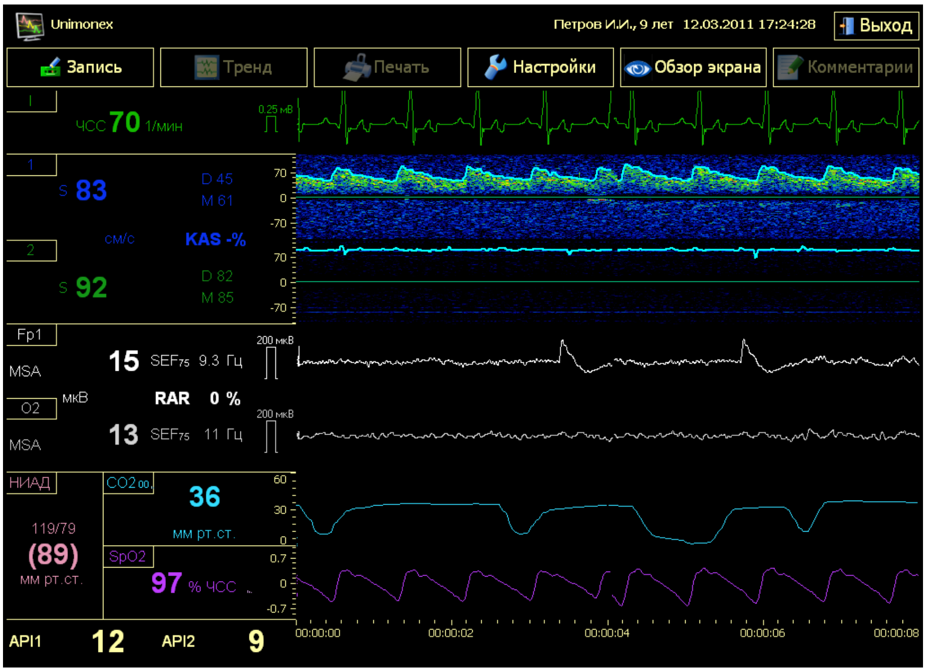

Рисунок 1.2 - Робоче вікно Unimonex в режимі on-line реєстрації даних

Unimonex призначений для інтеграції в одному сеансі дослідження різноманітних засобів і методів діагностики фізіології пацієнта з метою реалізації комплексних методів дослідження дисфункцій організму. Ефективність його застосування визначається можливістю одночасного дослідження відразу декількох систем організму, що дозволяє підвищити якість і достовірність висновку і, як наслідок, забезпечити високу ефективність лікування. Кількість інтегрованих методик і їх призначення визначається клінічними аспектами досліджень і може варіюватися в широких межах. Unimonex дозволяє вирішити основну проблему інтеграції даних - синхронізувати реєстровані при обстеженні потоки даних, представлені в різних форматах, в єдину систему комплексного аналізу і обробки діагностичної інформації. Одночасно можуть виконуватися: електрокардіографія, доплерографія (Рис 1.3), стабілометра, електроенцефалографія, пульсоксиметрія, Капнографії, електроміографія; моніторинг системного тиску; моніторинг інвазивного тиску.

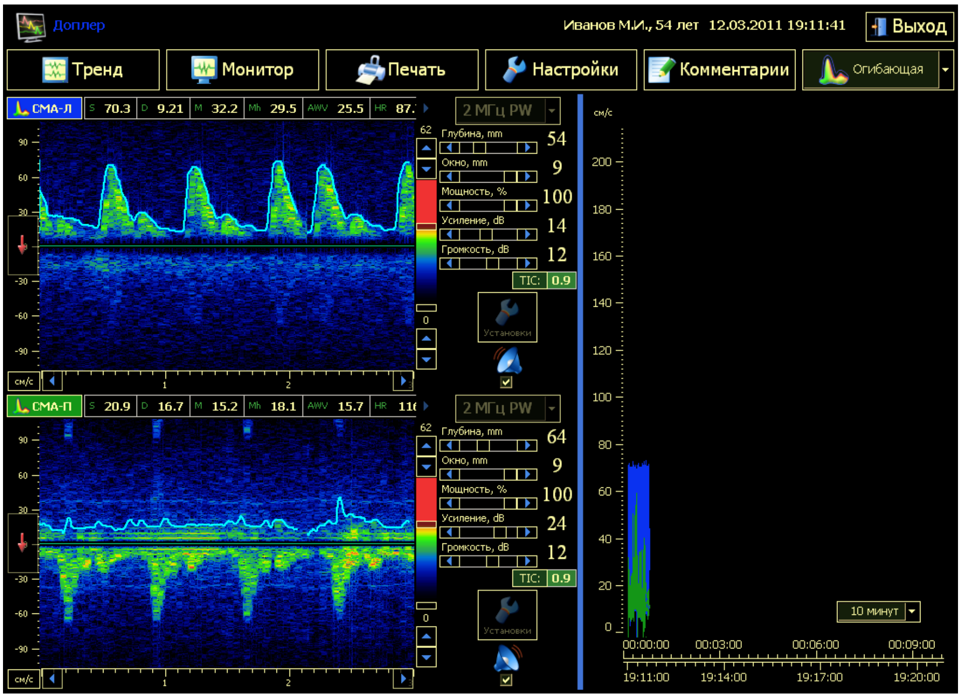

Рисунок 1.3 - Робоче вікно доплерографічної методики дослідження

Комплекс має розвинені засоби налаштування складу і кількості використовуваних при обстеженні методик і приладових модулів.

Багатоканальна система досліджень Unimonex дозволяє об'єднувати різне діагностичне обладнання в рамках одного сеансу досліджень. Синхронізація інформаційних потоків здійснюється спеціальними технічними і програмними засобами. Вибір спільно використовуваних методик визначається завданнями клінічних досліджень конкретного виду патології. Візуалізація даних кожного з каналів має два формати представлення: стислий і розгорнутий. Перший, використовується для відображення інформації одночасно по всіх каналах діагностики, другий - для деталізації даних конкретної методики. Стиснутий формат уявлення має наступну структуру: в лівій частині каналу відображається панель керування(Рис 1.4) з розрахунковими індексами, в правій - графік реєстрованого сигналу або тренд розрахункового індексу. Така форма подання найбільш ефективна для візуального сприйняття і аналізу декількох потоків інформації і добре зарекомендувала себе на практиці. Інформаційна панель одночасно з індикацією виконує функцію переходу до розгорнутій формі подання даних. Для цього досить клацнути по ній мишею і буде відкрито вікно відповідної методики досліджень. Таким чином, поєднуючи дві форми подання інформації, Unimonex дає користувачеві

розвинені засоби навігації в/ інформаційному полі багатовимірної діагностичної системи реєстрації параметрів стану пацієнта, забезпечує гнучкість аналізу і ефективність діагностики в цілому.

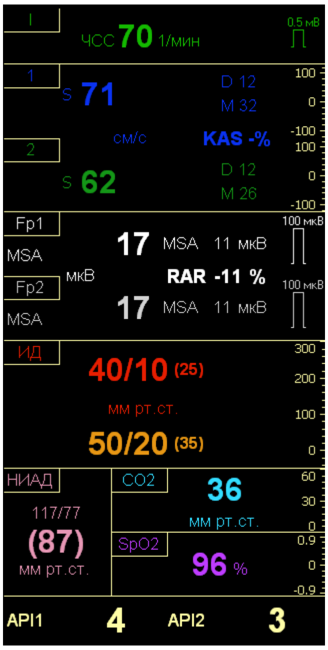

Рисунок 1.4 – Панель керування Unimonex

1.3 Smart пристрої та додатки для прийому медикаментів

Одна з проблем сучасної охорони здоров'я, і не тільки в нашій країні, полягає в тому, що ми постійно не виконуємо призначення лікарів, забуваємо приймати запропоновані нам ліки. З забудькуватістю людей боротися складно, але можна придумати методи, що дозволяють її подолати.

Дослідження показують, що порушення правил при прийомі ліків обходиться урядам країн Європи приблизно в 125 млрд € і сприяє передчасної смерті майже 200 000 європейців щорічно. У Росії статистика стверджує, що 37% людей, яким лікарі виписують ліки, які не дотримуються режиму їх прийому або перестають їх приймати завчасно. Питання дотримання доз і термінів прийому ліків відповідно до призначення лікаря і дотримання всієї призначеної терапії мають часто вирішальне значення для отримання сприятливого результату лікування.

Існує кілька інноваційних методів, які мають великий потенціал, щоб зробити значний вплив на виправлення ситуації. Ці методи стали широко використовувати в розвинених країнах.

Нагадування в смартфоні. Сьогодні майже всі мають мобільні телефони, найближчим часом їх замінять смартфони. Наприклад, в США їх вже більше, ніж персональних комп'ютерів. І спеціалізоване додатки або сервіс нагадувань про те, що пора прийняти ті або інші ліки. Дослідження, опубліковані в Journal of Medical Internet Research продемонстрували, що мобільні додатки допомагають поліпшити прийом ліків навіть для літніх людей, які не мають досвіду спілкування зі смартфоном, комп'ютером або Інтернетом.

"Розумні" контейнери для ліків, підключені до Мережі. Ці пристрої контролюють кількість таблеток в контейнері і можуть відправляти текстові або телефонні нагадування пацієнтам про необхідність приймати ліки. Деякі дослідження показали, що ці пристрої до 95% підвищують ступінь дотримання термінів прийому ліків.

Нижче приведено декілька smart-рішень

GlowCap - це спеціальна кришка, яка підходить до більшості баночок з ліками і починає блимати помаранчевим світлом, коли прийшов час прийняти ліки. Потім через годину, якщо ви її не послухалися, GlowCap починає програвати мелодію. Вона поставляється разом каганцем, який знаходиться на бездротового зв'язку з кришкою і також починає блимати в потрібний час. Що знаходиться всередині кришки чіп визначає, коли відкривається баночка і відправляє відповідне повідомлення по бездротовій мережі в додаток, що дозволяє лікарю (або родичам) контролювати наскільки точно виконується режим прийому ліків.

Рисунок – 1.5 спеціальна smart-кришка

Якщо людина все одно забуде прийняти ліки після нагадування, то він через пару годин отримає додаткове повідомлення на свій телефон.

У підставі кришки є кнопка, натиснувши на яку, ви надсилаєте через мережу AT & T автоматичний запит в найближчу аптеку, з проханням поповнити запас закінчилися таблеток. При цьому передбачається обов'язковий зворотний дзвінок з підтвердженням прийняття замовлення. Залишається тільки зайти в аптеку і забрати свої ліки.

Баночка для ліків AdhereTech виконує аналогічну функцію - сигналізує пацієнту, коли йому треба прийняти свої ліки і стежить за регулярністю споживання і дозуванням. Це забезпечується шляхом бездротового підключення до хмарного сервісу та збору даних про використання таблеток.

Рисунок 1.6 – Баночка для ліків AdhereTech

Баночка AdhereTech нагадує про необхідність прийняти таблетку шляхом дзвінка або текстового повідомлення на телефон, а також миготінням світла прямо в баночці. Вона містить чіп бездротового зв'язку CDMA, який автоматично відправляє невелика кількість даних, про те, коли пляшка була відкрита, і скільки таблеток (або рідкого ліки) було прийнято, що важливо, оскільки люди похилого віку можуть плутати дозування. Для вимірювання дозування і вологості використовується вбудований в банку сенсор. Це поки єдине на ринку засіб, що визначає, скільки таблеток використовував пацієнт.

В цілому AdhereTech використовується як звичайна баночка з ліками. Всі настройки, включаючи дані дозування ліків, знаходяться в хмарі, а пацієнт отримує

вже готовий до використання продукт. Система однаково добре працює з провідний і мобільним зв'язком.

Баночка може працювати 45 днів без підзарядки. Ідея виробника полягала в тому, щоб ця баночка продавалася в аптеці разом з ліками, а після місяця використання замінювалося на нову, повністю заряджену і забезпечену комплектом ліків на місяць. Компанія-виробник планує працювати в партнерстві з фармацевтичними компаніями, особливо по відношенню до дорогих ліків, щоб гарантувати використання їх пацієнтами.

В епоху мобільних технологій розробники розробили безліч програм, призначених для допомоги людям, хто забуває про приймати такі необхідні їм ліки.

Pill Reminder - це додаток, що працює на iPhone, що не має ніякого апаратного підкріплення. Це не тільки легка у використанні і гнучка система нагадування про необхідність прийняти ліки або поповнити аптечку, але і зручний засіб ведення особистої медичної картки.

Рисунок 1.7 – Додаток Pill Reminder

Інтуїтивно зрозумілий інтерфейс дозволяє, використовуючи вбудовану базу даних ліків, дуже швидко створити список своїх ліків і графік нагадувань. При створенні нагадувань можна вводити додаткові коментарі, наприклад, "після їжі",

"при такому-то рівні болю" і т.п., а також фотографії ліків, щоб не плутати їх. Звукові сигнали нагадувань можуть варіюватися по мелодії і тривалості.

Всі дані знаходяться тільки на смартфоні і нікуди не передаються. При цьому існує можливість експорту даних для передачі інформації, наприклад, лікаря.

Одне з особливостей програми - це інтеграція з великою базою даних лікарських засобів, де міститься вся найновішу інформацію про багатьох медикаментах і яку підтримує сам виробник програми. І, додаючи в свою систему нові ліки, ви можете миттєво дізнатися про його дозування, взаємодії з іншими ліками, побічні ефекти - і все це прямо в додатку.

Dosecast - це ще один додаток, тепер вже не тільки для пристроїв на базі iOS, але і Android, яке призначене для допомоги людям в своєчасному прийомі ліків і вітамінів, а також для контролю всього процесу медикаментозного лікування.

Рисунок 1.7 – Додаток Dosecast

Програма дозволяє легко заповнити список ваших ліків і поставити їм у відповідність потрібну кількість нагадувань. При цьому Dosecast має широкий набір звукових сигналів і варіантів їх використання: повтор сигналів через певний час, тривалий сигнал до тих пір, поки ви не почуєте і не відключіть його, облік тимчасових зон при подорожі та інше. Челове може створювати розпорядок прийому ліків на

день, тиждень або місяць, може встановлювати максимальну кількість доз на день, щоб уникнути передозування. Dosecast відстежує, коли людина спить або коли починається / закінчується курс лікування.

1.4 Мобільні технології в охороні здоров'я

Концепція мобільних технологій «m - Health» в охороні здоров'я набагато ширше, ніж просте використання малогабаритних медичних приладів, прикріплених до людського тіла. Під «m - Health» зазвичай розуміють сервіси, послуги, програми, заходи і будь-які інші дії в області медицини і охорони здоров'я, при реалізації яких застосовуються мобільні пристрої і інші технології бездротового зв'язку. «M - Health» - застосування мобільних технологій в сфері медичних послуг, а також інформації в галузі охорони здоров'я. На даний момент в світі мобільна медицина повсюдно використовується в лікуванні, а також попередженні інфекційних і неінфекційних захворювань, інформування вагітних жінок, турботі про людей похилого віку, боротьбі з дитячою та материнською смертністю. На підставі звітів компанії J'son & Partners Consulting, швидко розвивається сектором ринку «m - Health» є мобільні додатки для смартфонів в області охорони здоров'я. Виділяють 4 основних категорії мобільних додатків: загальне здоров'я і фітнес, медична інформація, віддалене консультування і спостереження, управління здоров'ям.

На підставі оцінок учасників ринку мобільна медицина буде інтенсивно рости. Лікарі, пацієнти найбільше у великих містах будуть прагнути до надання та отримання медичних послуг із застосуванням мобільних рішень. В області «m - Health» проводяться типи послуг з різноманітними по складності технологіями, так само і складні додатки для мобільних телефонів, з вбудованими можливостями або належними пристроями знімають життєві показники - тиск, рівень цукру та інше. Фахівці компанії J'son & Partners Consulting виділяють наступні види послуг, що реалізуються в рамках «m - Health» з різними за складністю технологіями:

- проекти з застосуванням SMS розсилок для інформування, організаторами яких можуть бути держава, громадські компанії, фармацевтичні компанії та інші;
- мобільна телемедицина;
- завантажувані додатки в Інтернет

- магазинах, які застосовуються користувачами в різних областях;

- складні додатки для мобільних телефонів, з вбудованими можливостями або належними пристроями, що знімають життєві показники

- тиск, рівень цукру та інші.

У звіті компанії «MarketsandMarkets» - «Mobile Health Apps & Solutions Market by Connected Devices, Health Apps, Medical Apps Global Trends & Forecast to 2018», наведені дані про те, що mHealth - додатки, призначені для турботи про здоров'я (wellness + fitness) нагромаджує 90% завантажень на ринку мобільного охорони здоров'я. Слід наголосити, що «m - Health» - це не тільки прогресивні технології, це абсолютно новий підхід взаємодії пацієнта з лікарем і догляду за хворими. Фахівці різних компаній сходяться в тому, що в найближчі 5 років має сенс чекати виникнення ефективних додатків «m - Health», спеціалізованих для віддаленого моніторингу та консультацій. Розмір ринку мобільних додатків в області охорони здоров'я зібрав трохи більше 2,4 мільярда доларів США в 2013 р, між тим уже до завершення 2017 року цей показник, згідно з прогнозами, збільшиться на 61% і перевищить 26 мільярдів доларів відповідно відомостями відомості «mHealth App Developer Economics 2014 »аналітичної фірми« Research2Guidance ». У зведенні ж фірми «MarketsandMar - kets» - «Mobile Health Apps & Solutions Market by Connected Devices, Health Apps, Medical Apps - Global Trends & Forecast to 2018», наведені трохи інші значення і розповідається про те, що розмір ринку мобільного охорони здоров'я в 2013 р зібрав 6,34 млрд. дол. і передбачається, що до 2018 року його середньорічний ритм підйому складе 26,7%, що дозволить досягти 20,68 млрд. дол.

1.5 Методи діагностики

Хвороби серця у останні десятиліття вийшли в 1-ий план. Наука далеко не стоїть на місці, з кожним роком з'являються нові способи діагностики і лікування, що можуть допомогти боротися з хворобами різної етіології. Кардіологія завжди була однією з найбільш значущих медичних наук. Проходить безперервна «боротьба» з хворобами серцево - судинної системи. На зміну вже давно поширених способів діагностики і лікування приходять нові. Ефективним прикладом, може служити дослідження мікроальтернацій ЕКГ, який дає можливість передбачити початок

серцево - судинної патології. Встановлено, що серце вважається свого роду самостійною системою, у якій є особиста «електростанція» - ділянки, в яких виникають нервові імпульси, що змушують серцеві стінки скорочуватися. Але яким би незалежним не було серце, на нього впливає і нервова система, як симпатична, так і парасимпатична, яка може послужити причиною до перебоїв в роботі серця. Одним з нинішніх способів оцінки зв'язку серця і нервової системи є оцінка варіабельності серцевого ритму (ВСР).

«Варіабельність» - це така якість біологічних процесів, яке пов'язане з потребою пристрою організму до мінливих обставин довкілля. Іншими словами, варіабельність - це нестійкість різних характеристик, в тому числі і ритму серця, у відповідь на вплив будь - яких обставин. Таким чином, варіабельність серцевого ритму (ВСР) відображає роботу серцево - судинної системи і роботу механізмів регуляції цілого організму. Вченими був виявлений взаємозв'язок між вегетативною нервовою системою і смертністю від серцево - судинних захворювань, в тому числі непередбачену загибель.

Варіабельність серцевого ритму представляє з себе більш практичний показник, внаслідок якого можливо оцінити результативність взаємодії серцево - судинної та інших систем організму. Цей аналіз стає поширеним внаслідок своєї простоти, так як вважається не інвазивних. Дане дослідження починають стрімко застосовувати в багатофункціональної діагностиці, так як показник варіабельності серцевого ритму дає можливість надати загальну оцінку про стан пацієнта, так як відображають життєво важливі характеристики управління фізіологічними функціями організму, до них зараховують багатофункціональні резерви механізмів його управління і вегетативний баланс.

Вплив симпатичного нерва веде до підвищення частоти серцевих скорочень за рахунок стимуляції бета - адренорецепторів синусового вузла. У свою чергу блукаючий нерв спонукає холінорецептори синусового вузла і проводить до брадикардії. Симпатична система має найбільший вплив над шлуночками, в той час як блукаючий нерв впливає на синусовий та атріовентрикулярна вузли.

На частоту серцевих скорочень впливають фази дихання. Під час вдиху пригнічується вагусний вплив (вплив блукаючого нерва) і прискорюється ритм. Під час видиху серцева діяльність сповільнюється, так як дратується блукаючий нерв. Можна відзначити, що серцевий ритм вважається відповіддю організму на вплив подразників внутрішнього і зовнішнього середовища. Таким чином, зміна ритму стане зустрічній реакцією на зміну будь - яких умов і регулюється симпатичним і парасимпатичних відділами нервової системи.

Сердечно - судинна система - приклад оригінальної системи управління, побудованої відповідно до ієрархічним принципом, де будь-який нижній рівень в стандартних обставин діє самостійно. При змінах зовнішнього середовища і / або при формуванні патологічного процесу з метою заощадження гомеостазу активізуються вищі рівні управління. Процес пристосування вимагає витрачання інформативних, енергетичних і метаболічних ресурсів організму. Керівництво ресурсами залежить з пред'явлених до організму умов зовнішнього середовища і здійснюється через нервові, ендокринні, гуморальні механізми, які щодо можна розбити на автономні і центральні. Втручання центральних механізмів управління в роботу автономних відбувається тільки в тому випадку, коли останні припиняють оптимально здійснювати свої завдання.

Аналіз варіабельності серцевого ритму

Останнім часом великої популярності в кардіологічних дослідженнях набирає аналіз варіабельності серцевого ритму, який базується на знаходженні черговості інтервалів R - R електрокардіограми. Ще називають NN - інтервали (normal - to - normal), тобто передбачаються проміжки тільки між нормальними скороченнями.

Завдяки даному аналізу можна отримати зведення про вплив на роботу серця вегетативної нервової системи і ладу гуморальних і рефлекторних факторів.

Аналіз варіабельності серцевого ритму надає можливість оцінити функціональний стан людини, крім цього дозволяє спостерігати за динамікою і виявляти патологічні стани. Дає можливість отримати зведення про адаптаційних резервах організму, що надає можливість передбачити збої в роботі серцево - судинної системи.

Зниження параметрів вказує на порушення взаємодії вегетативної нервової та серцево - судинної системи і проводить до патологій, зв'язаних з роботою серця. Більш високі показники варіабельності серцевого ритму властиві для здорових молодих людей і спортсменів, так як для них характерний більш високий парасимпатичний тонус. Різного роду захворювання серця органічної природи ведуть до зменшення показників варіабельності - високий симпатичний тонус. Раптове зниження - висока ймовірність летального результату.

У нинішній час є ряд способів оцінки варіабельності серцевого ритму. З числа них акцентують 3 категорії:

- методи швидкоплинної області - покладаються в статистичні методи і орієнтовані на дослідження сукупної варіабельності;
- методи частотної області - дослідження періодичних складових ВСР;
- інтегральні показники ВСР (зараховують Автокорреляционная дослідження і кореляційний ритмографії).

1.6 Статистичні методи

Дані методи застосовуються з метою прямої чисельної оцінки ВРС в досліджуваний період часу. При їх застосуванні кардіоінтервалограмми, показана на рисунку 1.5, розглядається як комплекс послідовних часових проміжків - інтервалів RR. Статистичні властивості динамічного ряду кардіоінтервалів містять: SDNN, RMSSD, PNN5O, CV.

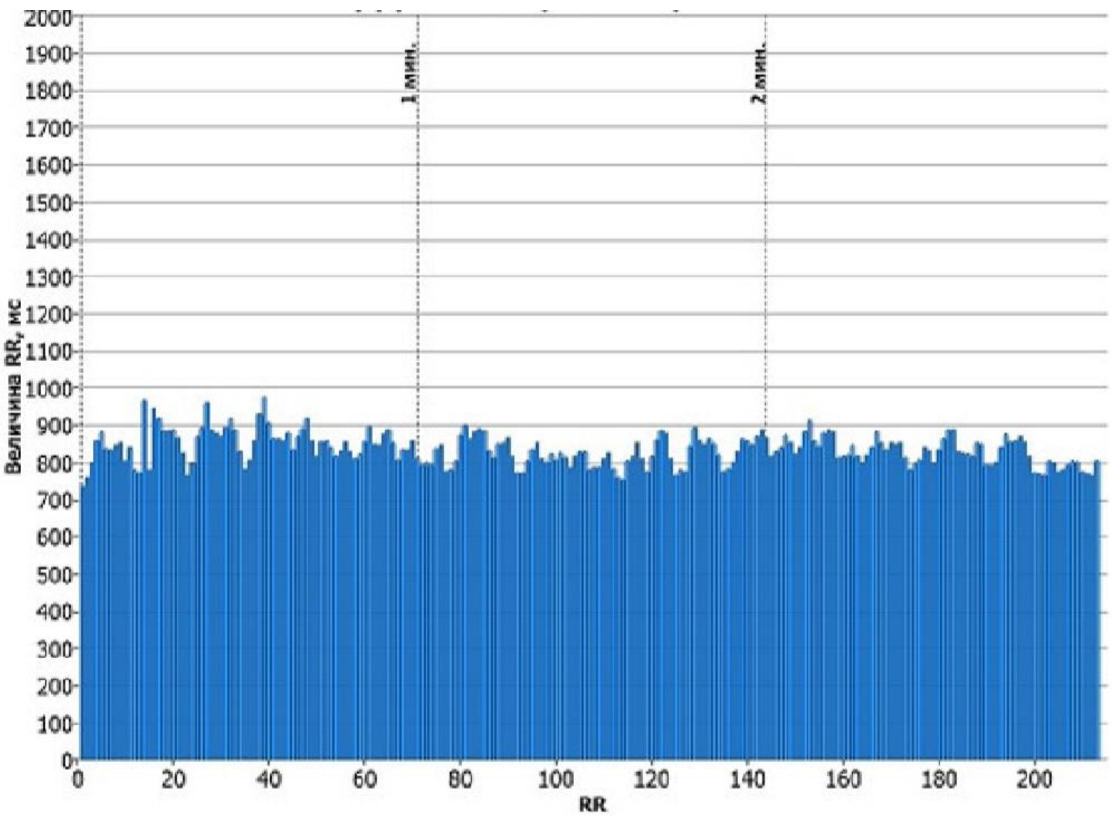

Рисунок 1.5 - Формування кардіоінтервалограмми (КІГ)

- SDNN або СКО - загальний показник варіабельності величин інтервалів RR за весь аналізований проміжок (NN - означає ряд нормальних інтервалів «normal to normal» з виключенням екстрасистол);

- СКО - середньоквадратичне відхилення (виражається в мс.);

- SDNN - стандартне відхилення NN інтервалів (подобу СКО);

- SDANN - стандартне відхилення середніх значень SDNN з 5 хвилинних частин для записів середньої тривалості, тривалих або 24 - х годинних записів. Таким же способом можуть відзначатися і стандартні відхилення середніх значень інших показників;

- RMSSD - квадратний корінь з суми квадратів різниці величин послідовних пар інтервалів NN (нормальних інтервалів RR);

- NN5O - кількість пар послідовних інтервалів NN, що відрізняються більш, ніж на 50 мілісекунд, отримане за повний проміжок записи;

- PNN5O (%) - відсоток NN50 від загальної кількості послідовних пар інтервалів, що відрізняються більш, ніж на 50 мілісекунд, отримане за повний проміжок записи;

- CV - коефіцієнт варіації. Він зручний для практичного застосування, так як передбачає собою нормовану оцінку СКО;

- CV = СКО / M * 100, де M - середнє значення інтервалів RR;

- D, As, Ex - 2-ий, 3-ий і 4-ий статистичні моменти. D - це СКО в квадраті, відображає підсумкову потужність всіх періодичних і неперіодичних коливань. As - коефіцієнт ассіметріі дає можливість оцінювати про стаціонарності досліджуваного динамічного ряду, про присутність і вираженості перехідних процесів, в тому числі трендів. Ex - коефіцієнт ексцессівності відображає темп зміни випадкових нестаціонарних компонентів динамічного ряду і відображає присутність локальних нестаціонарних.

1.7 Геометричні методи (варіаційна пульсометрія)

Суть варіаційної пульсометрії полягає в дослідженні закону розподілу кардіоінтервалів як випадкових величин. При цьому створюється варіаційна крива (крива розподілу кардіоінтервалів – гістограма) і формуються її головні властивості: Mo (Мода), Амо (амплітуда моди), MxDMn (варіаційний розмах). Мода-це найбільш часто трапляється в цьому динамічному ряді значення кардіоінтервалу. При нормальному розподілі і високої стаціонарності досліджуваного процесу Mo мало відрізняється від математичного очікування (M). Амо- (амплітуда моди) – це число кардіоінтервалів, відповідних значенню моди, в % до розміру вибірки. Варіаційний розмах (MxDMn) відображає рівень варіативності значень кардіоінтервалів в досліджуваному динамічному ряду. Він обчислюється згідно різниці максимального (Mx) і мінімального (Mn) значень кардіоінтервалів і тому при аритміях або артефактах може бути спотворений.

При побудові гістограм (або варіаційних пульсограмм) основне значення має підбір методу угруповання даних. У багаторічній практиці сформувався класичний підхід до сортування кардіоінтервалів в спектрі від 400 до 1300 мс. з інтервалом в 50 мс. Подібним способом, виділяються 20 фіксованих діапазонів тривалостей кардіоінтервалів, що дає можливість зіставляти варіаційні пульсограми, придбані різними дослідниками на різних групах досліджень. При цьому розмір добірки, в якій виконується угруповання і створення варіаційної пульсограми, також стандартний –

5 хвилин. Інший метод побудови варіаційних пульсограмм полягає в тому, щоб спочатку встановити модальне значення кардіоінтервалу, а потім, застосовуючи діапазони по 50 мс, створювати гістограму в обидві сторони з моди. Варіаційна пульсограма може бути також показана "гладким" графіком щільності розподілу, представлений на рисунку 1.6.

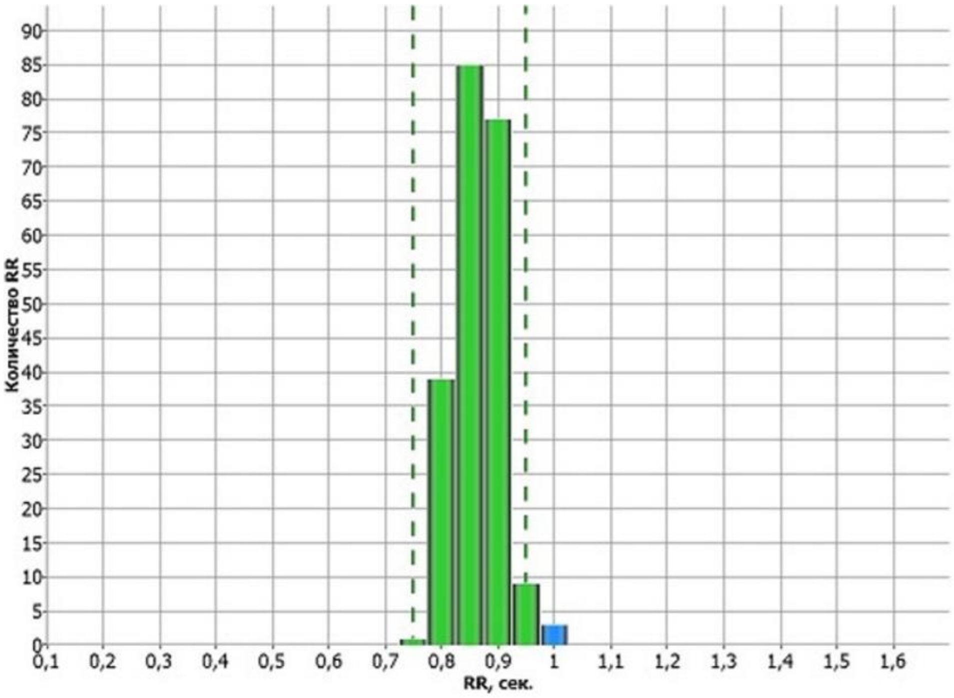

Рисунок 1.6 - Зразок варіаційної пульсограми

Західноєвропейські та американські вчені застосовують апроксимацію кривої розподілу кардіоінтервалів трикутником і обчислюють так званий тріангулярний показник – Інтеграл щільності розподілу (єдине число кардіоінтервалів) віднесений до максимуму щільності розподілу (AMo). Даний показник позначається як TIN (triangular interpolation of NN intervals). Крім цього, застосовується створення гістограм згідно різницевих значень розташованих поруч кардіоінтервалів з апроксимацією їх експоненціальної кривої і обчисленням логарифмічного коефіцієнта, а також інші методи апроксимації.

1.8 Автокореляційний аналіз

Розрахунок і створення автокореляційної функції динамічного ряду кардіоінтервалів орієнтоване на дослідження внутрішньої структури даного ряду як випадкового процесу. Автокореляційна функція являє собою графік динаміки коефіцієнтів кореляції, одержуваних при послідовному зміщенні розглянутого динамічного ряду на одне число по відношенню до свого власного ряду.

Після першого зсуву на одне значення коефіцієнт кореляції тим менше одиниці, чим більше проявлені дихальні хвилі, представлені вгорі на малюнку 1.5. Якщо в досліджуваній вибірці переважають повільнохвильові елементи, то коефіцієнт кореляції після першого зсуву стане лише несуттєво нижче одиниці, представлені в середині і внизу на рисунку 1.7. Подальші зрушення ведуть до поступового скорочення кореляційних коефіцієнтів.

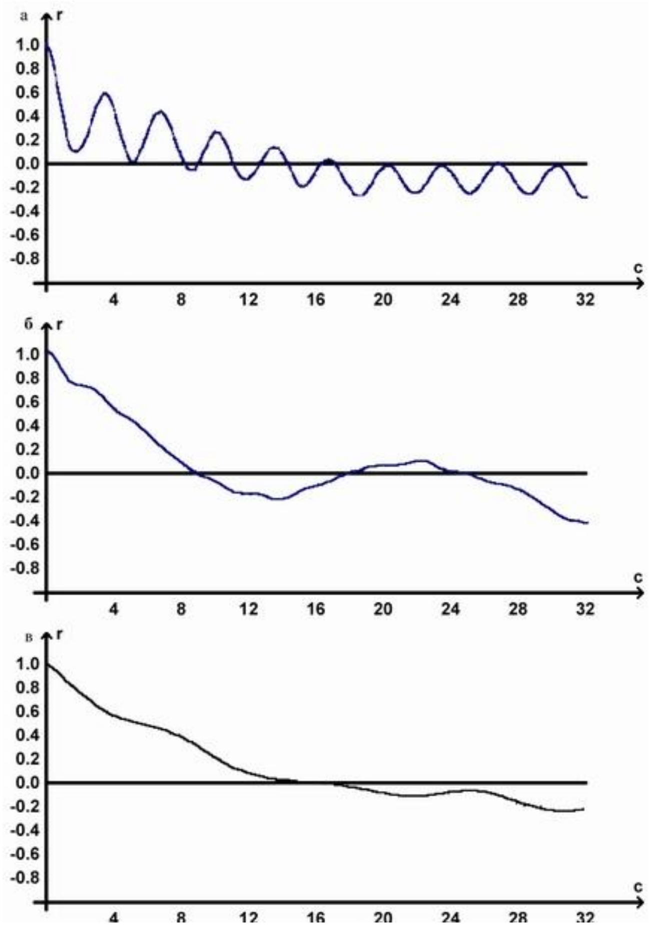

Рисунок 1.7 - Зразки автокоррелограмм з вираженими дихальними хвилями (вгорі), з переважанням повільних (в середині) і дуже повільних (внизу) хвиль.

В якості кількісних характеристик автокорелограми рекомендуються С1-значення коефіцієнта кореляції після першого зсуву і С0-число зрушень, в слідстві якого значення коефіцієнта кореляції ставати негативним.

Суть способу кореляційної ритмографії полягає в графічному відображенні послідовних пар кардіоінтервалів (попереднього і наступного) в двомірній координатній площині. При цьому по осі абсцис відкладається значення R-Rn, а по осі ординат – значення R – Rn+1. Графік і область точок, придбаних таким чином (плями Пуанкаре або Лоренца), називається кореляційної ритмограммой або скаттерограммой (scatter – розвіювання). Даний метод оцінки ВСР відноситься до способів нелінійного розгляду і вважається особливо корисним для випадків, коли на тлі монотонності ритму трапляються унікальні і несподівані патології (ектопічні зменшення і (або) «випадання» окремих серцевих скорочень).

При побудові скаттерограми утворюється комплекс точок, центр яких знаходиться на бісектрисі. Відстань від середини до початку осей координат відповідає більш прогнозованої тривалості серцевого циклу (Мо). Розмір відхилення точки від бісектриси в ліву сторону демонструє, в якій мірі цей серцевий цикл коротше попереднього, в праву сторону з бісектриси – в якій мірі він триваліше попереднього. Передбачається визначати наступні показники скаттерограммы:

- довжина головного «без екстрасистол і реліквій) «Хмари» (тривала вісь еліпса – L) відповідає варіаційному розмаху. За фізіологічним змістом даний показник не виділяється від SDNN, тобто відображає загальний результат регуляції ВСР, але показує на найбільшу амплітуду коливань тривалості інтервалів R-R;

- ширина скаттерограми (перпендикуляр до тривалої осі, проведений через її центр-w);

- площа скаттерограми розраховується згідно з формулою площі еліпса:

$$S = \pi ab$$

Нормальна форма скаттерограмми передбачає собою овал, витягнутий уздовж бісектриси. Безпосередньо таке розміщення еліпса означає, що до дихальної додана якась величина недихательной аритмії. Форма скаттерограмми у вигляді кола означає відсутність недихательних компонентів аритмії. Вузький овал, представлений на

малюнку 1.8, відповідає переважанню недихательних компонентів в загальній варіабельності ритму, яка формується протяжністю «Хмари» (скаттерограми).

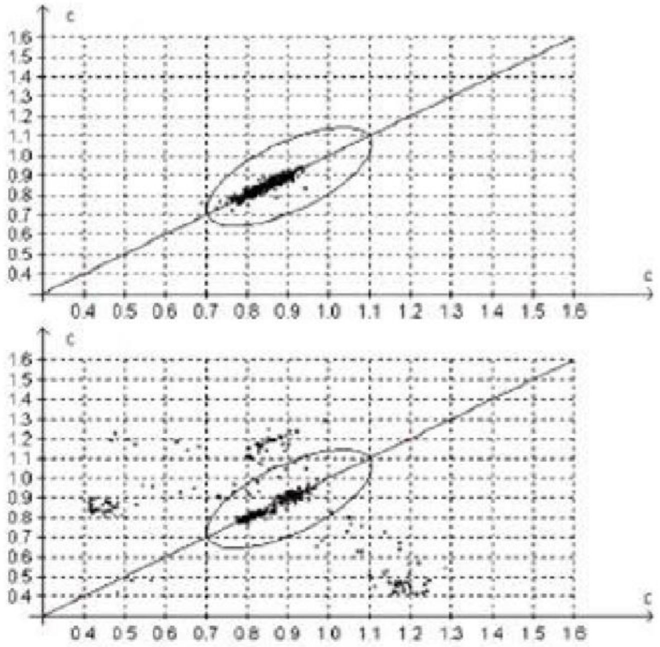

Рисунок 1.8 - Зразки кореляційних ритмограм (КРГ) - скатерограм, вгорі- нормальна КПГ, внизу - у пацієнта з аритмією

Протяжність еліпса добре корелювала з розміром HF, а ширина з LF, представлені на рисунку 1.8. При аритміях, коли способи статистичного і спектрального аналізу варіабельності серцевого ритму малоінформативні або неприйнятні, раціонально застосовувати оцінку кореляційної ритмограми.

1.9 Спектральні методи аналізу ВСР

Спектральні способи аналізу ВСР придбали в нинішній час досить широке поширення. Дослідження спектральної щільності потужності коливань надає інформацію про розподіл потужності в залежності від частоти коливань. Використання спектрального аналізу дає можливість кількісно дати оцінку різні частотні складові коливань ритму серця і чітко графічно показати співвідношення різних компонентів СР, що відображають динамічність конкретних ланок регуляторного механізму.

Розрізняють параметричні і непараметричні способи спектрального аналізу. До первинних відноситься авторегресійний аналіз, до інших – швидке перетворення

Фур'є (БПФ) і періодограмний аналіз. Обидві дані групи методів дають порівнянні результати.

Параметричні, і зокрема авторегресійні, методи вимагають відповідності розглянутого предмета конкретним моделям. Єдиним для всіх традиційних способів спектрального аналізу вважається проблема використання функції вікна (Windowing). Головне призначення вікна-зниження величини зміщення в періодограммних спектральних оцінках. Є конкретні відмінності спектрального оцінювання даних при застосуванні періодограмного методу з однорідним вікном (при 256 значеннях RR) і використанні різноманітних рівнів міжсегментного зсуву і різноманітного числа відліків на сегмент.

Збільшення дозволу при зростанні міжсегментного зсуву і числа відліків на сегмент тягне за собою виникнення маси додаткових піків в Спектрі і підвищення амплітуди піків в правій середині спектра. При спектральному аналізі ВСР істотне значення має розмір аналізованої вибірки. При коротких записах (5 хвилин) виділяють три основних спектральних елементи. Ці компоненти відповідають діапазонам дихальних хвиль і повільних хвиль 1 – го і 2 – го порядку, представлені на рисунку 1.9).

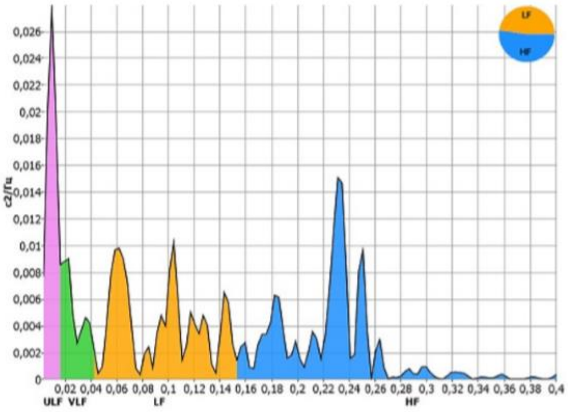

Рисунок 1.9 - Типова спектрограма ВСР при використанні методу швидкого перетворення Фур'є (БПФ)

У західній літературі відповідні спектральні елементи набули назви високочастотних (High Frequency – HF), низькочастотних (Low Frequency – LF) і дуже низькочастотних (Very Low Frequency – VLF).

Частотні діапазони кожного з трьох вищевказаних спектральних елемента вважаються дискусійними. За євро-американськими рекомендаціями пропонуються наступні діапазони частот:

- високочастотний діапазон – дихальні хвилі)–0,4–0,15 Гц (2,5-6,5 сек);

- низькочастотний діапазон (повільні хвилі 1-го порядку) - 0,15-0,04 Гц (6,5-25 сек);

- дуже низькочастотний діапазон (повільні хвилі 2 – го порядку) – 0,04 -0,003 Гц (25 – 333 сек).

При розгляді тривалих записів виділяють також ще й ультра низькочастотний елемент-Ultra Low Frequency (ULF) з частотами більше 0,003 Гц.

Досвід вітчизняних досліджень та підсумки досліджень, проведених багатьма іноземними авторами, демонструють необхідність корекції цих рекомендацій. Це належить головним чином до діапазону VLF.

Пропонується подальша скоригована модель частотних діапазонів при спектральному аналізі ВСР: пропоноване лімітування діапазону VLF до 0,015 Гц обумовлено тим, що при аналізі 5 – хвилинних записів ми фактично грунтовно можемо встановлювати тільки коливання з періодом в 3 – 4 рази меншим, ніж тривалість реєстрації сигналів (тобто близько 1 – ї хвилини). Тому пропонується всі коливання з періодом більше хвилини визначати до діапазону ULF і вже в ньому акцентувати належні піддіапазони. При спектральному аналізі зазвичай для кожного з компонентів обчислюють абсолютну підсумкову потужність в діапазоні, середню потужність в діапазоні, значення найбільшої гармоніки і умовне значення у відсотках від підсумкової потужності у всіх діапазонах (Total Power – TP). При даному TP формується як сукупність потужностей в діапазонах HF, LF і VLF. Згідно з відомостями спектрального аналізу серцевого ритму розраховуються наступні характеристики: Індекс централізації-ІЦ (Index of centralization, IC = (HF+LF)/VLF) і Індекс вагосимпатичної взаємодії LF/HF.

Висновки по першому розділу. У розділі були проаналізовані мобільні технології в галузі охорони здоров'я, вивчені основні принципи варіабельності

серцевого ритму, методи аналізу та оцінка результатів аналізу. Також було розглянуто кілька медичних систем моніторингу.

2. ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Призначення та ціль системи

Проектування системи є основним завданням будь-якого проекту. Необхідно знайти оптимальне рішення, яке буде відповідати всім аспектам сучасної індустрії. Для вирішення поставленого завдання були складені технічні, в якому перераховані вимоги до функціональних і нефункціональних характеристик розроблюваної системи. Проектування включає в себе добірку найбільш підходящої мови програмування, бази даних, середовище розробки, так як це є невід'ємною частиною проектування системи. У разі неправильного вибору присутня ймовірність зіткнення з низкою проблем. Тому це є найважливішою частиною проекту.

Автоматизована система дистанційного моніторингу функціонального стану людини призначена для вирішення наступних завдань:

- збір і обробка варіабельності серцевого ритму людини за допомогою пульсометра;
- передача і зберігання даних для їх подальшого аналізу;
- висновок про стан здоров'я людини.

Основними цілями створення автоматизованої системи дистанційного моніторингу функціонального стану людини є:

- дистанційне визначення поточного функціонального стану здоров'я;
- тривале спостереження за функціональним станом здоров'я;
- самоконтроль пацієнтів з метою реалізації зворотного зв'язку.

2.2 Структура системи моніторгнгу

Моніторні системи виконують такі функції:

- забезпечення моніторингу безпеки пацієнтів в реанімаційних блоках,
- забезпечення проведення діагностики амбулаторних пацієнтів,
- використання пацієнтами домашнього монітора для віддаленого вра-чебного спостереження,
- збір всіх даних моніторингу і діагностики в базу даних на цен-тральних сервері,

- забезпечення віддаленого доступу до бази даних в межах лікарні та з інших лікарень. Всі окремі сучасні моніторингові засоби різних виробників мають структуру представлену на рисунку 2.1

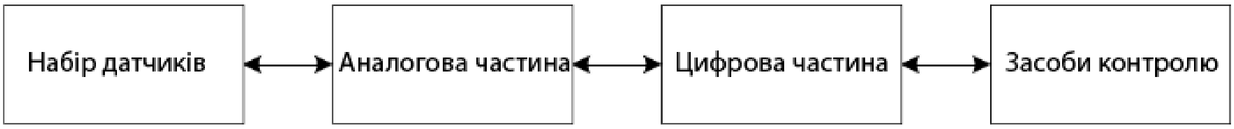

Рисунок 2.1 - Структура моніторингової апаратури

Набір датчиків призначений для перетворення деяких фізичних величин організму людини в електричні сигнали. Аналогова частина призначені для отримання, посилення, фільтрації і предобработки інформації про фізичних величинах організму людини. Цифрова частина призначена для обробки, індикації, зберігання та обміну інформацією про фізичних величинах організму людини і результатів їх обробки. Засоби контролю призначені для індикації результатів моніторингу, а також управління процесом моніторингу.

Аналогову частину практично всі виробники проектують і виготовляють самостійно, або компонують її використовуючи підсилювальні мо-дулі різних сторонніх виробників. Цифрову частину реалізують двома шляхами:

- ведення власної розробки і виготовлення,
- використання стандартних комп'ютерів.

Власну розробку і виготовлення ведуть підприємства мають свою виробничу базу, яку потрібно забезпечити роботою. При цьому термін розробки медичної техніки в цілому зростає, а розроблені в результаті пристрою є, по суті, спеціалізованими комп'ютерами. Ці спеціалізовані комп'ютери, які складно модифікувати під інші завдання. При появі нового завдання її цифрову частину, як правило, розробляють заново. До того ж, потрібно приділяти увагу дизайну отриманих виробів. Перевагою даного походу є можливість компоновки аналогової і цифрової частини в єдиному корпусі. Даний підхід реалізують в основному великі виробники обчислювальної техніки, для яких медична техніка є однією з серії виробів (Hiewlett Packard, Hellige).

У другому підході використовуються різні стандартні обчислювальні засоби. До недавнього часу вони обмежувалися стандартними персо-нальними комп'ютерами (ПК) (IBM-сумісними), ноутбуками, різними промисловими комп'ютерами або одноплатними комп'ютерами на різних платформах. Стандартні настільні комп'ютери знаходили обмежене застосування в зв'язку з великими весогабарітнимі параметрами для стаціонарного застосування. Ноутбуки та промислові комп'ютери мають висо-кую вартість. Одноплатні комп'ютери вимагають розробки обрамлення і виготовлення корпусу.

2.3 Об'єкт управління та опис систем моніторингу

Моніторингова система призначена для комплексного інформаційно-аналітичного забезпечення процесів:

- фіксування пацієнтів в БД;
- складання історії про курси лікування і прийомах;
- реєстрація всіх процесів;
- занесення даних в історію пацієнта.

Основними цілями створення системи моніторингу пацієнта є:

Заміщення існуючої інформаційної системи на повністю автоматизовану, що значно спростить витрати часу і сил на роботу з пацієнтами.

Дозволить підвищити оперативність роботи з клієнтами. Забезпечить можливість надання послуг більшій кількості пацієнтів.

Підвищить якість роботи з клієнтами. Зменшить ризик помилок при редагуванні і додаванні інформації про пацієнтів.

Наочність і структурність даних забезпечить зручність в роботі з інформацією.

Поєднуючи в собі переваги існуючих систем телемедицини та телеметрії, віддалений моніторинг має переваги, що дозволяють синтезувати «інформаційний портрет» хворого (групи пацієнтів) в режимі реального часу, не прив'язуючись до його фізичного місцезнаходженням. При цьому реалізуються такі унікальні можливості:

Висока швидкість передачі даних. Реальний трафік високошвидкісного широкосмугового Internet досягає 100 Мб / сек, що дозволить забезпечити стійку

передачу електрофізіологічних показників стану пацієнтів спеціалізованим службам медичних закладів в режимі реального часу.

Комплексування показників. У медичний заклад одночасно надходять потоки даних про стан відразу декількох пацієнтів. Реєстровані дані проходять три рівня комплексування. На першому рівні об'єднуються дані з електрофізіологічного устаткування по кожному пацієнту, на другому рівні - дані декількох пацієнтів і на третьому - дані декількох медичних установ, де розміщуються пацієнти. Для комплексування потоків медичної інформації використовуються апаратно-програмні засоби ,що пройшли апробацію при вирішенні аналогічних завдань віддаленого зв'язку.

Ведення бази даних моніторингу. Всі потоки даних, що надходять в організацію по віддаленій зв'язку, автоматично зберігаються в медичній базі даних і строго асоціюються з медичними картами спостережуваних пацієнтів. Це дозволяє фахівцям реалізувати Постпроцессорний аналіз зареєстрованої інформації з можливістю її роздруківки, архівування та експорту на зовнішні носії. Таким чином, по кожному «віддаленого» пацієнтові лікаря завжди буде доступний повний обсяг нативной інформації про динаміку зміни показників хворого за весь час спостереження.

Розвинений сервіс on-line аналізу вхідних потоків інформації. У процесі спостереження віддалених пацієнтів фахівцям надаються можливості коментування (текст, аудіо запис голосу лікаря) реєструється інформації в будь-який момент часу, виконання оперативних роздруківок окремих ділянок записи, реалізації операцій спеціалізованої обробки даних і ряд інших можливостей, що підвищують ефективність on-line аналізу. Отже, візуальне сприйняття динаміки електрофізіологічних показників доповнюється широким функціональним сервісом її обробки.

Безперервне, цілодобовий моніторинг. Стан пацієнта реєструється у вигляді сукупності електрофізіологічних показників з частотою оцифрування не менше 100 Гц. Сумарний потік інформації при одночасному спостереженні 15 пацієнтів становить до 1 Мб / сек. Всі дані цілодобово надходять на Інтернет - шлюзи медичної

організації і розподіляються по відділах, відображаючись на терміналах для оперативного аналізу.

Синхронізація всіх потоків в режимі реального часу. Всі потоки даних, що реєструються з «віддалених» пацієнтів в режимі реального часу, строго синхронізовані. Всі події записуються в пам'ять системи автоматично і коментуються фахівцями в on-line режимі. Такий підхід дозволяє отримати якісну оцінку клінічної ситуації по кожному пацієнту.

2.3.1 Характеристика об'єкта автоматизації

Об'єктом автоматизації є процеси моніторингу життєвих показників людини, аналізу даних, виявлення ризику розвитку контрольованих захворювань. Автоматизовані процеси дистанційної системи включають в себе:

- моніторинг показників здоров'я: збір, зберігання, обробка;
- аналіз даних життєвих показників з метою виявлення ризику їх погіршення;
- зберігання даних в базі даних;
- процеси супроводу відновного лікування;
- медичний самоконтроль. Процес дистанційного моніторингу включає в себе збір і інтерпретацію результатів вимірювань пульсометра:
- для спостереження здорових пацієнтів в профілактичних цілях;
- для спостереження пацієнтів, що проходять курс лікування,
- для спостереження пацієнтів похилого віку.

2.3.2 Вимоги до структури та функціонування системи

Автоматизована система дистанційного моніторингу функціонального стану людини повинна бути децентралізованою, тобто всі дані повинні розташовуватися в центральному сховищі.

Автоматизована система дистанційного моніторингу функціонального стану людини повинна мати трирівневу архітектуру.

В системі пропонується виділити наступні функціональні підсистеми:

- підсистема збору, обробки і завантаження даних, яка призначена для реалізації процесів збору даних з систем джерел, приведення зазначених даних до виду, необхідного для наповнення підсистеми зберігання даних;

- підсистема зберігання даних, яка призначена для зберігання даних в структурах, націлених на прийняття рішень;

- підсистема формування та візуалізації звітності, яка призначена для формування бізнес-орієнтованих вітрин даних та звітності.

Як протокол взаємодії між компонентами системи на транспортно-мережевому рівні необхідно використовувати протокол HTTP.

2.3.3 Питання які вирішує система моніторингу

Дані спочатку записуються в електронному вигляді, що створює передумови для переходу на електронний документообіг позбавляючи від паперового дублювання

Обмін даними в електронному вигляді відбувається дуже швидко (відстань не має принципового значення)

Автоматизовані системи в цілодобовому режимі приймають дані і виробляють їх обробку, що позбавляє від необхідності в умовах обмеженого часу провести прийом хворих, в той же час сприяючи попередженню розвитку захворювань на ранній стадії

Створює умови при яких один фахівець може обслужити більшу кількість пацієнтів, без безпосередньої присутності пацієнта

З огляду на названих причин знижуються витрати на надання медичних послуг і потреба в кадрах, підвищується зручність і доступність отримання медичних послуг пацієнтами

2.4 Інформаційна модель та її опис

Процес проектування інформаційної системи є довготривалою і трудовитратною роботою, що вимагає повного розуміння всіх аспектів системи. Для якісної реалізації здійснюється структурний підхід, на основі якого проводиться систематизація і стандартизація всіх процесів розробки програмного забезпечення.

Принцип структурного підходу до проектування інформаційної системи полягає в її декомпозиції на функціональні блоки.

За допомогою програмного продукту Bpwin і методології IDEF0 були змодельовані бізнес-процеси автоматизованої системи дистанційного моніторингу функціонального стану людини.

Для відображення призначення системи та її взаємодія із зовнішнім середовищем побудуємо контекстну діаграму засобами методології IDEF0. У кожній моделі може бути тільки одна контекстна діаграма. Після опису основної функції виконується функціональна декомпозиція, тобто визначаються функції виходять з основної.

На рисунку 2.2 показана контекстна діаграма «моніторинг стану здоров'я людини».

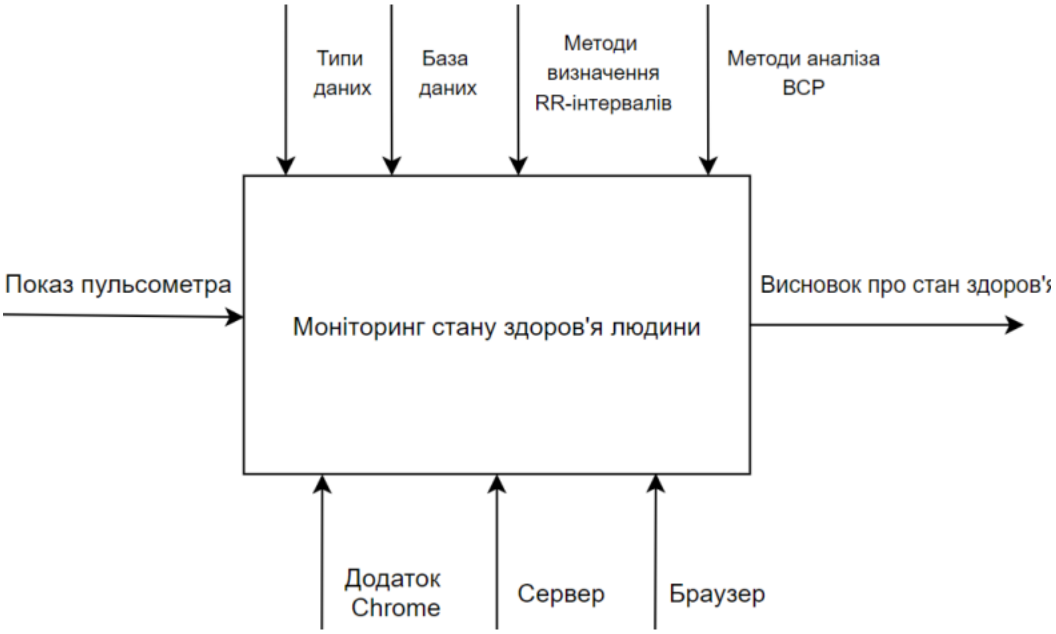

Рисунок – 2.2 Контекстна діаграма «Моніторинг стану здоров'я людини»

Контекстна діаграма «Моніторинг стану здоров'я людини» описує загальний процес моніторингу здоров'я. На вхід надходять дані пульсометра. Після ряду операцій, виконаних згідно з методами аналізу ВСР, пацієнт може ознайомитися з висновком про стан здоров'я. Головним джерелом обробки даних, що надходять від додаток Chrome, є сервер.

Провівши декомпозицію контекстної діаграми, виділимо процеси, що протікають під час моніторингу.

На рисунку 2.3 показана декомпозиція контекстної діаграми «моніторинг стану здоров'я людини».

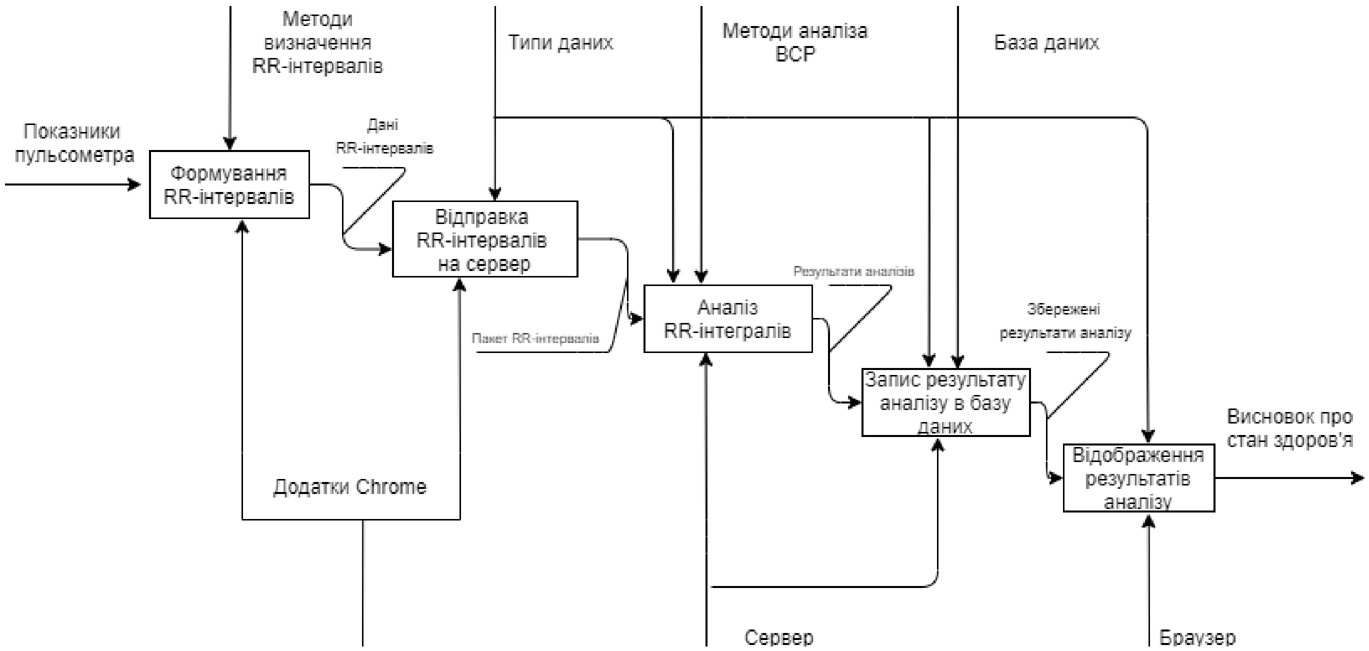

Рисунок 2.3 - Декомпозиція контекстної діаграми

Вхідними даними для реалізації моніторингу стану здоров'я людини є показання пульсометра у вигляді потоку бітів за допомогою послідовного порту.

Процес під номером 1, на рисунку 2.3, описує розшифровку потоку бітів в Rr інтервали за допомогою додаток Chrome.

Процес під номером 2, на рисунку 2.4, описує відправку готового пакету RR інтервалів на сервер.

Процес під номером 3, на рисунку 2.3, описує інтерпретацію сервером отриманих RR – інтервалів.

Процес під номером 4, на рисунку 2.3, описує запис в базу даних показань, отриманих в результаті проведення аналізів.

Процес під номером 5, на рисунку 2.3, є завершальним. Полягає в відображення результатів аналізів, отриманих з бази даних, попередньо оброблених.

На рисунку 2.4 показана декомпозиція другого рівня «формування RR інтервалів».

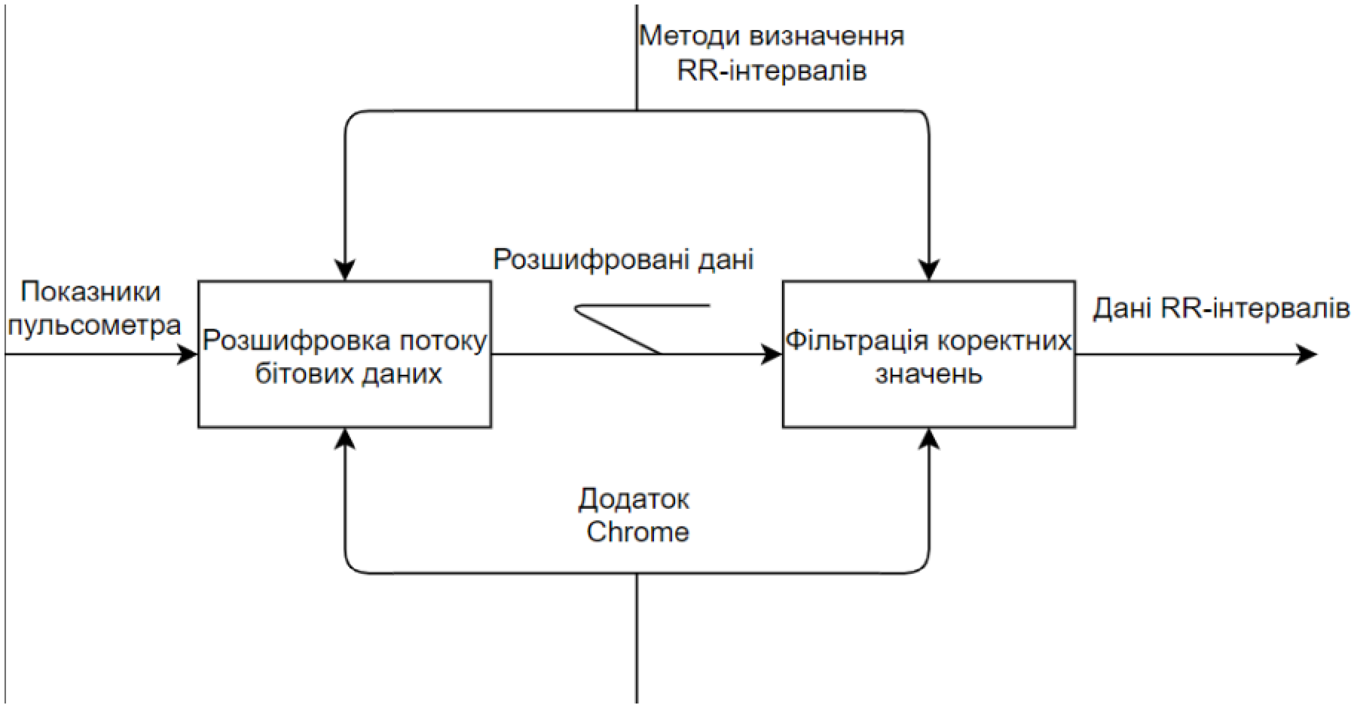

Рисунок 2.4 - Діаграма декомпозиції другого рівня «формування RR інтервалів»

При знятті показань пульсометр передає потік бітових даних за допомогою віртуального послідовного порту. додаток Chrome інтерпретує і фільтрує отримані дані, відповідно до методів визначення RR інтервалів.

Після фільтрації здійснюється відправка RR інтервалів на сервер.

На рисунку 2.5 показана декомпозиція другого рівня процесу «аналіз RR інтервалів».

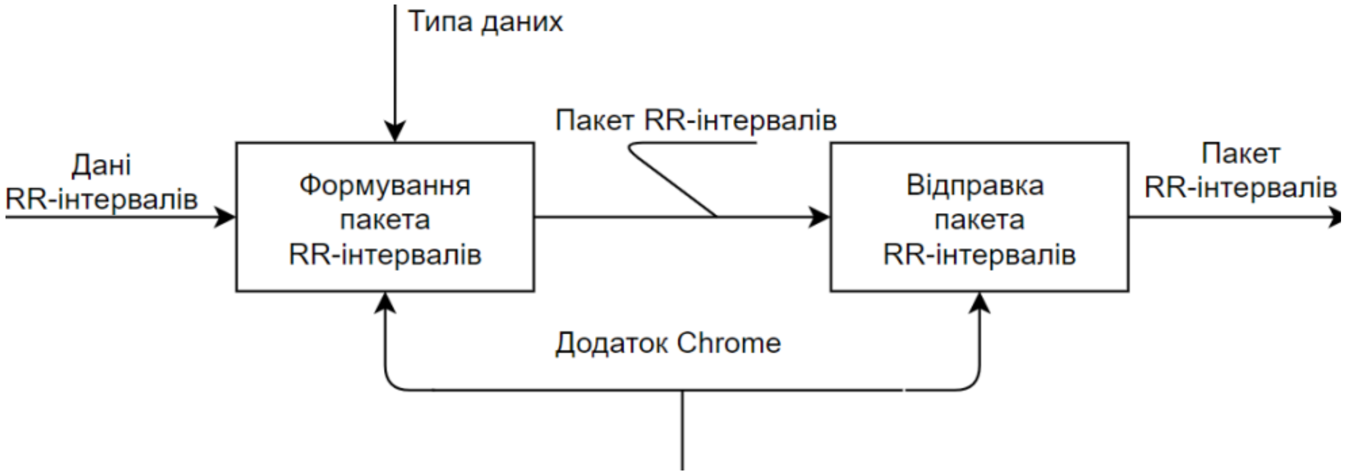

Рисунок 2.5 - Діаграма декомпозиції другого рівня " відправка пакету RR-інтервалів на сервер»

Перед відправкою на сервер, додаток Chrome формується пакет RR інтервалів згідно типізації даних.

Після відправки пакета даних, сервер аналізує отримані RR інтервали.

На рисунку 2.6 показана декомпозиція другого рівня процесу «аналіз RR інтервалів».

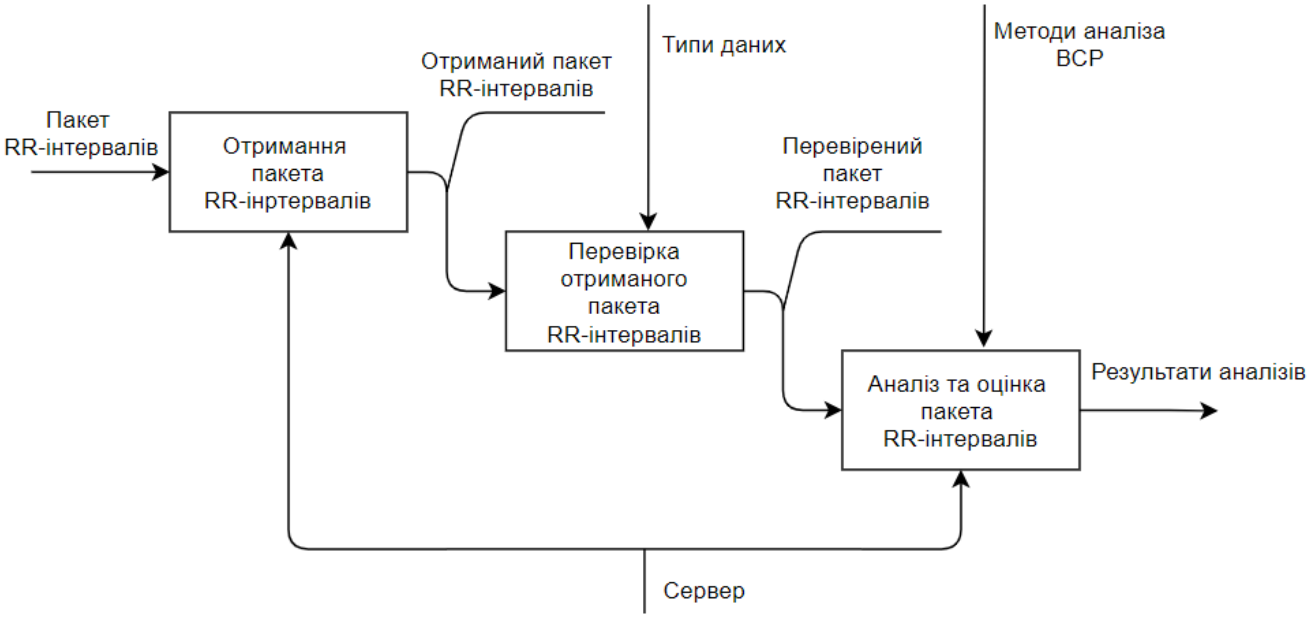

Рисунок 2.6 - Діаграма декомпозиції другого рівня " аналіз RR інтервалів»

Пакет RR інтервалів, отриманий сервером, проходить перевірку на відповідність типів даних. При успішній перевірці проводиться аналіз і оцінка результатів отриманих показань.

Отримані обчислення записуються в базу даних після аналізу.

На рисунку 2.7 показана декомпозиція другого рівня процесу «запис результатів аналізу в базу даних».

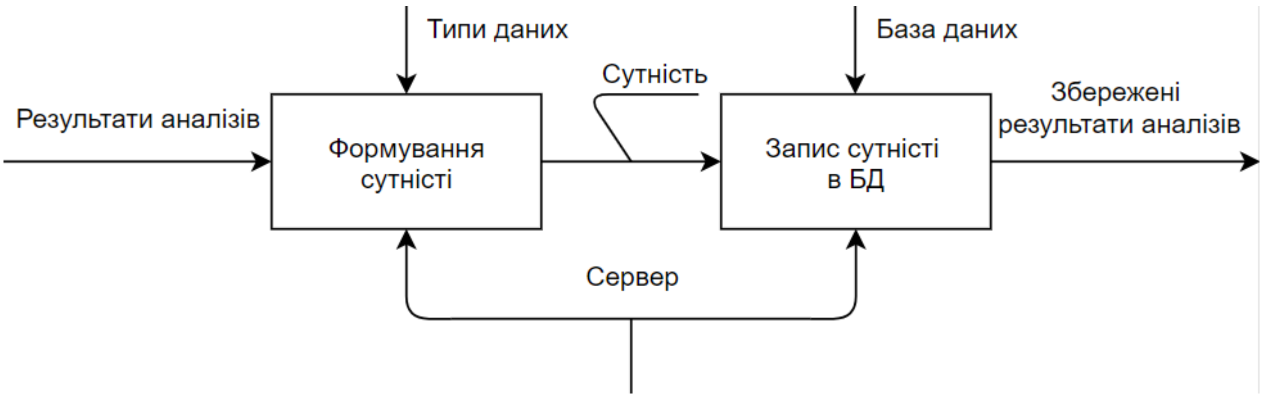

Рисунок 2.7 - Діаграма декомпозиції другого рівня «запис результатів аналізу в базу даних»

Для коректного запису в базу даних, необхідно перетворити дані в модель, відповідні її критеріям.

При подальшому зверненні браузера на сервер буде відображатися висновок про стан людини.

На рисунку 2.8 показана декомпозиція другого рівня процесу «відображення результатів аналізу».

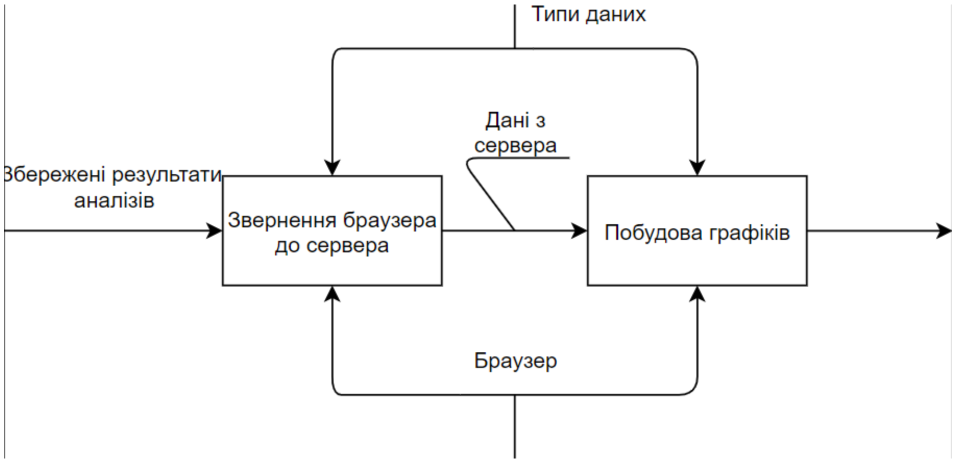

Рисунок 2.8 - Діаграма декомпозиції другого рівня «відображення результатів аналізу»

Фінальним етапом є відображення висновку про стан здоров'я. При запиті браузера на сервер, сервер віддає збережені результати аналізів RR інтервалів, браузер, оперуючи отриманими даними, будує графіки, тобто відображає висновок про стан здоров'я людини.

Браузер відправляє запит на сервер, який в свою чергу надсилає відповідь, останній в собі містить результати аналізів RR – інтервалів. Отримавши відповідь, браузер будує графіки на основі отриманих даних, що в своїй сутності є відображенням висновку про стан здоров'я людини.

2.5 Проектування бази даних

Для забезпечення наочності представлення інформації, яка буде зберігатися в базі даних застосовується інфологічне моделювання. Інфологічна модель даних забезпечує найбільш природним для людини способом збір і подання інформації про предметну область, яку передбачається зберігати в створюваній базі даних.

Ціль інфологічного моделювання - забезпечення найбільш дійсних для людини способів збору та представлення тієї інформації, яку зберігають в створюваній базі даних. Тому інфологічну модель даних намагаються створювати по аналогії з дійсною мовою. Основними конструктивними елементами інфологічних моделей є дійсності, зв'язки між ними та їх властивості (атрибути).

Дійсність - любий розрізняючий об'єкт (об'єкт, який ми можемо відрізнити від іншого), інформацію про який треба зберігати в базі даних. Дійсністю можуть бути люди, місця, смак, колір та ін. Необхідно розрізняти такі поняття як тип дійсності та екземпляр дійсності. Поняття тип дійсності відноситься до набору однорідних особистостей, предметів, подій або ідей, виступаючих як єдине ціле. Екземпляр дійсності відноситься до певної речі в наборі. Наприклад, типом дійсності може бути ім'я, а атрибутом - Сергій, Марина та ін.

Атрибут - поіменована характеристика дійсності. Його ім'я повинно бути унікальне для певного типу дійсності, але може бути однаковим для різних типів дійсності. Атрибути використовуються для визначення того, яка інформація повинна бути зібрана про дійсність. Прикладами атрибутів для дійсності автомобіль є тип, марка, номерний знак, колір. Тут також існують різноманіття між типом та екземпляром. Тип атрибуту колір має багато екземплярів чи значень: червоний, синій, зелений та ін., однак кожному екземпляру дійсності привласнюється тільки одне значення атрибуту.

Абсолютна різниця між типами дійсності та атрибутами відсутня. Атрибут є таким тільки в зв'язку з типом дійсності. В іншому контексті атрибут може виступати як самостійна дійсність. Наприклад, для автомобільного заводу колір - тільки атрибут продукту виробництва, а для лакофарбової фабрики колір - тип дійсності.

Ключ - мінімальний набір атрибутів, за значеннями яких можна однозначно знайти потрібний екземпляр дійсності. Мінімальність означає, що виключення із набору будь - якого атрибуту не дозволяє ідентифікувати дійсність по тим, що залишились. Для дійсності розклад ключем є атрибут номер_рейсу та пункт призначення.

На підставі аналізу предметної області виділимо сутності моделі і зобразимо їх у вигляді діаграми, яка показана на рисунку 2.9.

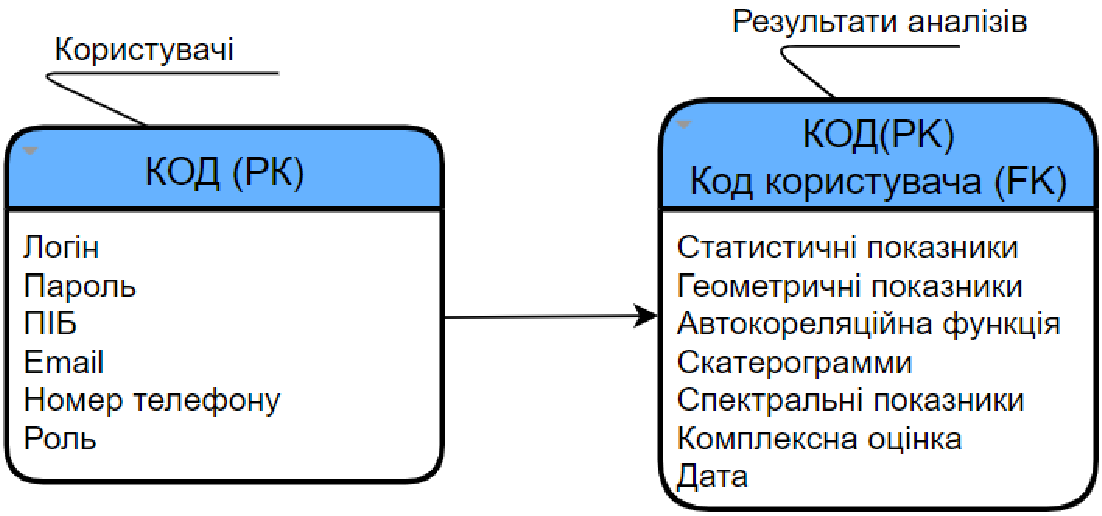

Рисунок 2.9 - Інфологічна модель бази даних

Проаналізуємо таблиці з інфологічної моделі.

Колекція "користувачі" зберігає в собі всі дані про користувачів системи: дані для авторизації, ПІБ, Email, номер телефону і роль (admin або user).

Колекція "результати аналізів" зберігає в собі результати досліджень, проведених на основі отриманих RR – інтервалів.

Висновки по другому розділу.

В даному розділі були визначені основні технічні вимоги: цілі створення системи, характеристика об'єкта автоматизації, вимоги до структури і функціонування системи.

Також були спроектовані і описані інформаційна модель системи та інфологічна модель бази даних.

3. РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Обґрунтування вибору технологій для розробки

Серверна частина організована за допомогою програмної платформи Node.js для мови програмування JavaScript.

3.1.1 Середовище розробки JavaScript

JavaScript (JS) — динамічна, об'єктно-орієнтована прототипна мова програмування. Реалізація стандарту ECMAScript. Найчастіше використовується для створення сценаріїв вебсторінок, що надає можливість на боці клієнта (пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд вебсторінки.

JavaScript класифікують як прототипну (підмножина об'єктно-орієнтованої), скриптову мову програмування з динамічною типізацією. Окрім прототипної, JavaScript також частково підтримує інші парадигми програмування (імперативну та частково функціональну) і деякі відповідні архітектурні властивості, зокрема: динамічна та слабка типізація, автоматичне керування пам'яттю, прототипне наслідування, функції як об'єкти першого класу.

Застосування

Мова JavaScript використовується для:

- написання сценаріїв вебсторінок для надання їм інтерактивності;
- створення односторінкових та прогерсивних вебзастосунків (React, AngularJS, Vue.js);
- програмування на боці сервера (Node.js(Express.js));
- стаціонарних застосунків (Electron, NW.js);
- мобільних застосунків (React Native, Cordova);
- сценаріїв в прикладних програмах (наприклад, в програмах зі складу Adobe Creative Suite чи Apache JMeter);
- всередині PDF-документів.

Незважаючи на схожість назв, мови Java та JavaScript є двома різними мовами, що мають відмінну семантику, хоча й мають схожі риси в стандартних бібліотеках та

правилах іменування. Синтаксис обох мов отриманий «у спадок» від мови С, але семантика та дизайн JavaScript є результатом впливу мов Self та Scheme.

JavaScript має низку властивостей об'єктно-орієнтованої мови, але завдяки концепції прототипів підтримка об'єктів в ній відрізняється від традиційних мов ООП. Крім того, JavaScript має кілька властивостей, притаманних функціональним мовам, — функції як об'єкти першого класу, об'єкти як списки, каррінг, анонімні функції, замикання (closures) — що додає мові додаткову гнучкість.

JavaScript має С-подібний синтаксис, але в порівнянні з мовою С має такі корінні відмінності:

- об'єкти, з можливістю інтроспекції і динамічної зміни типу через механізм прототипів;
- функції як об'єкти першого класу;
- обробка винятків;
- автоматичне приведення типів;
- автоматичне збирання сміття;
- анонімні функції.

JavaScript містить декілька вбудованих об'єктів: Global, Object, Error, Function, Array, String, Boolean, Number, Math, Date, RegExp. Крім того, JavaScript містить набір вбудованих операцій, які, грубо кажучи, не обов'язково є функціями або методами, а також набір вбудованих операторів, що управляють логікою виконання програм. Синтаксис JavaScript в основному відповідає синтаксису мови Java (тобто, зрештою, успадкований від С), але спрощений порівняно з ним, щоб зробити мову сценаріїв легкою для вивчення. Так, приміром, декларація змінної не містить її типу, властивості також не мають типів, а декларація функції може стояти в тексті програми після неї.

Сучасний JavaScript - це «безпечний» мову програмування. Він не надає низькорівневий доступ до пам'яті або процесору, тому що спочатку був створений для браузерів, які не потребують цього.

Можливості JavaScript сильно залежать від оточення, в якому він працює. Наприклад, Node.JS підтримує функції читання / запису довільних файлів, виконання мережевих запитів і т.д.

У браузері для JavaScript є все, що пов'язано з маніпулюванням веб-сторінками, взаємодією з користувачем і веб-сервером.

Наприклад, в браузері JavaScript може:

- Додавати новий HTML-код на сторінку, змінювати існуючий вміст, модифікувати стилі.
- Реагувати на дії користувача, клацання миші, перемістити вказівник, натискання клавіш.
- Відправляти мережеві запити на віддалені сервера, завантажувати і завантажувати файли (технології AJAX і COMET).
- Отримувати і встановлювати куки, задавати питання відвідувачеві, показувати повідомлення.
- Запам'ятовувати дані на стороні клієнта («local storage»).

JavaScript має ряд властивостей об'єктно-орієнтованої мови, але завдяки концепції прототипів підтримка об'єктів в ній відрізняється від традиційних мов ООП. Крім того, JavaScript має ряд властивостей, властивих функціональним мовам — функції як об'єкти першого класу, об'єкти як списки, каррінг, анонімні функції, замикання (closures) — що надає мові додаткову гнучкість.

JavaScript має C-подібний синтаксис, але в порівнянні з мовою C має наступні корінні відмінності:

- об'єкти, з можливістю інтроспекції та динамічної зміни типу через механізм прототипів;
- функції як об'єкти першого класу;
- Обробка винятків;
- автоматичне приведення типів;
- автоматичне збирання сміття;
- анонімні функції.

JavaScript містить кілька вбудованих об'єктів: Global Object, Error, Function, Array, String, Boolean, Number, Math, Date, RegExp. Крім того, JavaScript містить набір вбудованих операцій, які, грубо кажучи, не обов'язково є функціями або методами, а також набір вбудованих операторів, які керують логікою виконання програм. Синтаксис JavaScript в основному відповідає синтаксису мови Java (тобто, в кінцевому підсумку, успадкований від C), але спрощений в порівнянні з ним, щоб зробити мову сценаріїв легким для вивчення. Так, наприклад, декларація змінної не містить її типу, властивості також не мають типів, а декларація функції може стояти в тексті програми після неї.

3.1.2 Програмна платформа Node.js

Node.js — платформа з відкритим кодом для виконання високопродуктивних мережевих застосунків, написаних мовою JavaScript. Засновником платформи є Раян Дал (Ryan Dahl). Якщо раніше Javascript застосовувався для обробки даних в браузері користувача, то node.js надав можливість виконувати JavaScript-скрипти на сервері та відправляти користувачеві результат їхнього виконання. Платформа Node.js перетворила JavaScript на мову загального використання з великою спільнотою розробників.

Node.js має наступні властивості:

- асинхронна одно-нитева модель виконання запитів;
- неблокуючий ввід/вивід;
- система модулів Common JS;
- рушій JavaScript Google V8.
- Для керування модулями використовується пакетний менеджер npm (node package manager).

Платформа Node.js призначена для виконання високопродуктивних мережевих застосунків, написаних мовою програмування JavaScript. Платформа окрім роботи із серверними скриптами для веб-запитів, також використовується для створення клієнтських та серверних програм.

В платформі використовується розроблений компанією Google рушій V8.

Для забезпечення обробки великої кількості паралельних запитів у Node.js використовується асинхронна модель запуску коду, заснована на обробці подій в неблокуючому режимі та визначенні обробників зворотніх викликів (callback). Як способи мультиплексування з'єднань підтримується epoll, kqueue, /dev/poll і select. Для мультиплексування з'єднань використовується бібліотека libuv, для створення пулу нитей (thread pool) задіяна бібліотека libeio, для виконання DNS-запитів у неблокуючому режимі інтегрований c-ares. Всі системні виклики, що спричиняють блокування, виконуються всередині пулу нитей і потім, як і обробники сигналів, передають результат своєї роботи назад через неіменовані канали (pipe).

Клієнтська частина (додаток Chrome) також написана мовою програмування JavaScript, з використанням Google Chrome Extensions.

Google Chrome Extensions-API дозволяє збільшити функціонал браузера Chrome.

Для реалізації роботи з даними системи була обрана СУБД MongoDB.

MongoDB-кроссплатформенна документо-орієнтована система управління базами даних. Класифікована як база даних NoSQL,

MongoDB відходить від класичних основ реляційної структури бази даних на користь JSON-схожих документів з динамічними схемами, що здійснює інтеграцію даних в конкретних типах додатків швидший.

Основні особливості:

- документо-орієнтованість;
- спеціальні запити;
- індексація;
- реплікація;
- балансування навантаження;
- файл зберігання;
- агрегування;
- серверне здійснення JavaScript;
- блокування колекцій.

3.1.3 Середовище розробки WebStorm

JetBrains WebStorm — інтегроване середовище розробки для JavaScript, HTML та CSS від компанії JetBrains, розроблена на основі платформи IntelliJ IDEA. WebStorm є спеціалізованою версією PhpStorm, пропонуючи підмножину з його можливостей. WebStorm постачається з перед-установленим плаґінами JavaScript (такими як для Node.js), котрі доступні для PhpStorm безкоштовно.

WebStorm підтримує мови JavaScript, CoffeeScript, TypeScript та Dart.

WebStorm забезпечує автодоповнення, аналіз коду на льоту, навігацію по коду, рефакторинг, зневадження та інтеграцію з системами управління версіями. Важливою перевагою інтегрованого середовища розробки WebStorm є робота з проектами (у тому числі, рефакторинг коду JavaScript, що міститься в різних файлах і теках проекту, а також вкладеного в HTML). Підтримується множинна вкладеність (коли в документ на HTML вкладений скрипт на Javascript, в який вкладено інший код HTML, всередині якого вкладений Javascript) — в таких конструкціях підтримується коректний рефакторинг.

Використовуючи можливості коду HTML / XHTML і XML, WebStorm забезпечує автоматичне завершення стилів, посилань, атрибутів і інших елементів коду. При роботі з CSS здійснюється завершення коду класів, HTML-номерів, ключових слів і т. Д. WebStorm пропонує автоматичне рішення таких проблем, як вибір формату, властивостей, класів, посилань на файли і інших атрибутів CSS. Рішення дозволяє використовувати потужність інструменту Zen coding для верстки HTML, відображає дії тега на web-сторінці. Продукт WebStorm здійснює завершення коду JavaScript для ключових слів, лейблів, змінних, параметрів і функцій DOM і підтримує специфічні особливості популярних браузерів. Реалізовані в рішенні функції рефакторінга JavaScript дозволяють перетворювати структуру коду і файлів і .js.

WebStorm забезпечує налагодження коду JavaScript і надає широкий діапазон можливостей: знаходження точки зупину в HTML і JavaScript, настройка параметрів точки зупину, тестування синтаксису коду в режимі реального часу і т. Д. Продукт підтримує платформи JQuery, YUI, Prototype, DoJo, MooTools, Qooxdoo і Bindows.

WebStorm передбачає інтегровану перевірку тексту на теги, послідовність коду, помилки в написанні і т. Д. WebStorm дозволяє редагувати файли і автоматично синхронізувати їх на вимогу при віддаленій роботі або зберіганні.

Продукт підтримує функцію контролю версій і попередніх варіантів коду і фіксує всі вироблені дії та зміни. Завдяки створенню історії в WebStorm можна відновлювати кодові вирази, блоки і навіть цілі файли.

Основні можливості

- Інтеграція з системами управління версіями Subversion, Git, GitHub, Perforce, Mercurial, CVS підтримуються з коробки з можливістю побудови списку змін і відкладених змін;
- Інтеграція з системами відстеження помилок;
- Модифікація файлів .css, html, .js з одночасним переглядом результатів (Live Edit, в деяких джерелах ця функціональність називається «редагування файлів на льоту» або «в реальному часі» або «без перезавантаження сторінки»);
- Віддалене розгортання за протоколами FTP, SFTP, на монтованих мережевих дисках тощо з можливістю автоматичної синхронізації;
- Можливості Zen Coding і Emmet.

3.2 Опис програмного інтерфейсу системи

Серверна і клієнтська частина системи побудовані за архітектурою MVC, це означає, що модифікація будь-якого компонента може здійснюватися незалежно один від одного.

Архітектура MVC дозволяє нам розділити код програми на 3 частини: Модель (Model), Вид або Представлення (View) і Контролер (Controller). Вперше вона була описана в 1978 році, і призначалася для додатків з графічним інтерфейсом (віконцями і кнопками), але пізніше була адаптована і для веб-додатків.

Поділ на частини дозволяє спростити великий за обсягом код. Якщо код писати одним довгим скриптом, в ньому стає важко розібратися, і важко вносити зміни, не допустивши помилку.

MVC не прив'язана до якогось конкретного мови програмування, і не вимагає використання об'єктно-орієнтованого програмування або якийсь інший парадигми.

Серверний проект був побудований за допомогою пакетного менеджера – npm, який був розроблений спеціально для Node.js, для вирішення завдань з управління залежностями.

Взаємодія між компонентами MVC реалізується трохи по-різному в серверних і в десктопних додатках через те, що веб-додаток - короткоживучі, обробляє один запит користувача і завершується, а десктопних програм обробляє багато запитів без перезапуску.

У схемі з активною моделлю Вид підписується на зміни в моделі. Потім, коли відбувається якась подія (наприклад, користувач натискає кнопку), викликається контролер. Він дає моделі команду на зміну даних. Модель повідомляє своїм передплатникам (в тому числі увазі), що дані змінилися, і Вид оновлює інтерфейс програми. Ми не будемо далі розбирати цей варіант MVC, про нього докладно написано в моєму уроці по MVC в JS-додатках.

Схема з активною моделлю нагадує кільце або цикл, так як вона розрахована на тривалу роботу. На відміну від неї, схема з пасивною моделлю зазвичай використовується в короткоживучих додатках.

У серверних додатках використовується схема з пасивною моделлю. Припустимо, користувач заходить на сторінку форуму. Його браузер відправляє HTTP запит на отримання сторінки зі списком повідомлень. При цьому запускається контролер, який аналізує запит Користувача і запитує у моделі список повідомлень. Отримавши його, він викликає Вид і передає йому список, і той відображає його у вигляді веб-сторінки. Після цього скрипт завершується. Якщо Користувач захоче додати повідомлення, він заповнить форму, відправить її, викличеться контролер, який відповідає за обробку даних цієї форми, прийме дані, попросить Модель перевірити і вставити в базу даних нове повідомлення, і потім віддасть HTTP відповідь з редиректом на сторінку перегляду повідомлень.

У серверних додатках зазвичай використовується "пасивна" модель, а в десктопних додатках - "активна". Активна модель, на відміну від пасивної, дозволяє підписуватися і отримувати повідомлення про зміну в ній. У серверних додатках це не потрібно. На рисунку 3.1 наведена схема, яка зображує взаємодію компонентів:

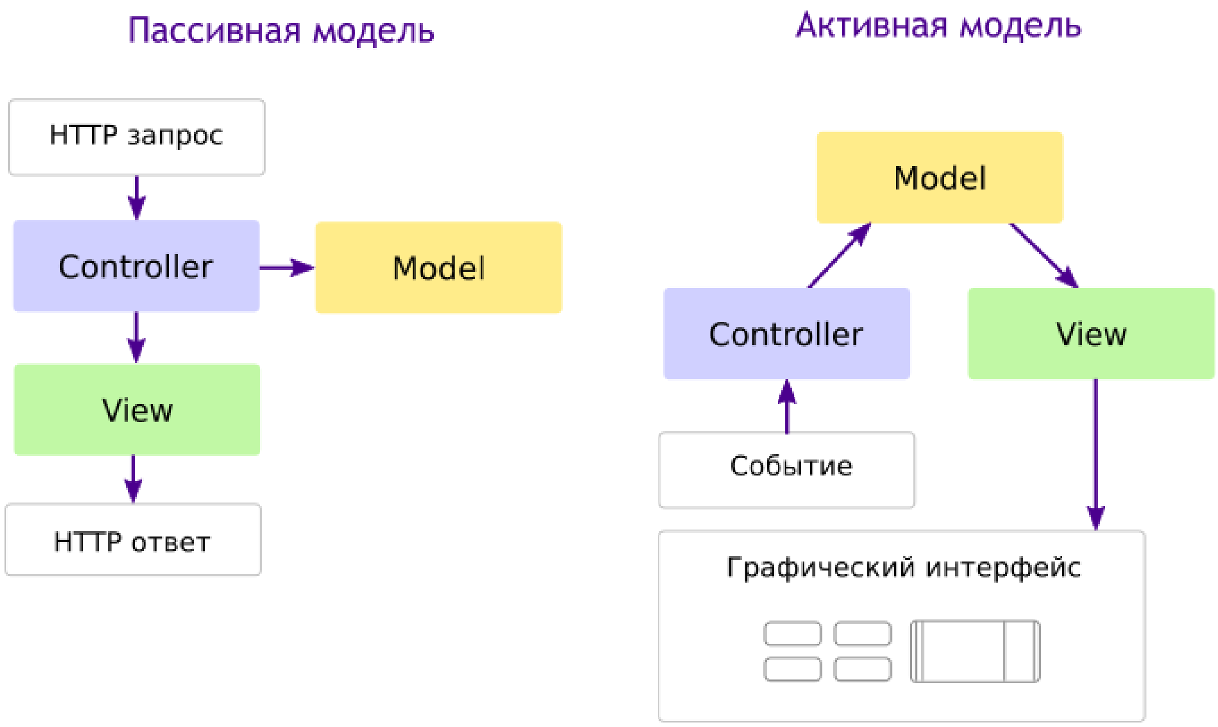

Рисунок 3.1 - Взаємодія компонентів MVC

Інтерфейс IDE і структура серверного проекту показана на рисунку 3.2.

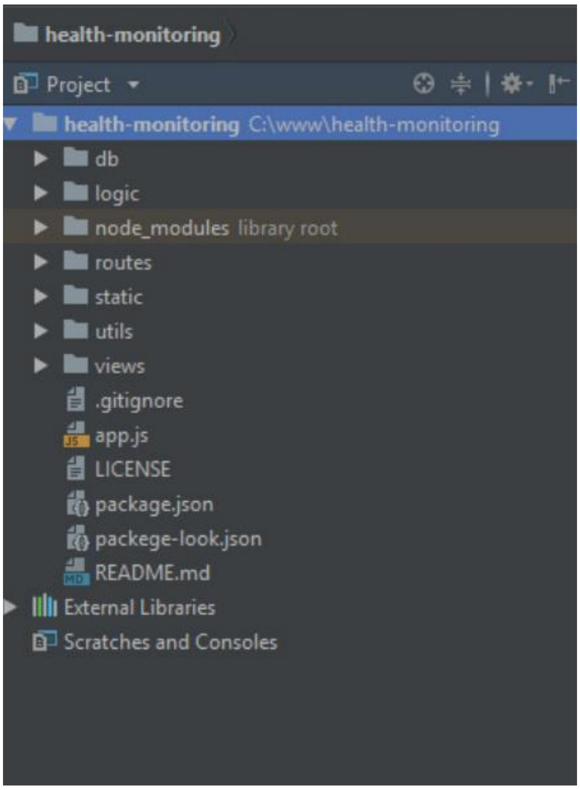

Рисунок 3.2 - Структура серверного проекту

Директорія «db» містить всі налаштування бази даних.

Директорія «logic» містить всі математичні функції методів аналізу варіабельності серцевого ритму.

Директорія «node_modules» містить всі бібліотеки, залежно яких вказані в файлі «package.json».

Директорія «routes» містить всі роутери сайту.

Директорія «static» містить всі файли і бібліотеки, які будуть використовуватися браузером.

Директорія «utils» містить всі допоміжні функції.

Директорія «views» містить всі шаблони сторінок.

Файл «app.js» є головним файлом додатки, в якому

відбувається його ініціалізація.

Файл «package-look.json» є файл контролю версій бібліотек, зазначених у файлі «package.json».

Клієнтський проект був створений за допомогою API Google Chrome- Chrome

Extensions, який дозволяє збільшити функціонал браузера Chrome.

Інтерфейс IDE і структура клієнтського проекту показана на малюнку 3.3.

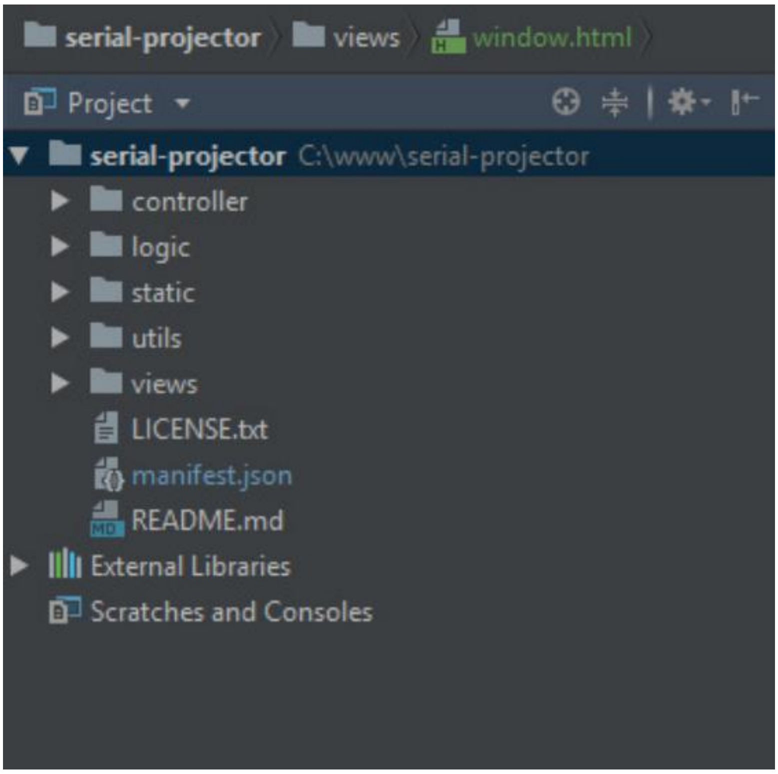

Рисунок 3.3 - Структура серверного проекта

Директорія «controller» містить всі основні функції програми.

Директорія «logic» містить всі функції розшифровки і фільтрації RR - інтервалів.

Директорія «static» містить всі файли і бібліотеки, які будуть використовуватися браузером.

Директорія «utils» містить всі допоміжні функції.

Директорія «views» містить всі шаблони сторінок.

Файл «manifest.json» є основним фалом настройки додаток.

3.3 Опис контрольного прикладу реалізації

Після проектування програмного забезпечення необхідно виконати контрольний приклад реалізації проекту. Нижче наведені докладні описи роботи системи.

Початком роботи в автоматизованій системі моніторингу функціонального стану людини є перехід за адресою сайту.

Головна сторінка системи представлена на рисунку 3.4.

Рисунок 3.4 – Головна сторінка

Кожен користувач зобов'язаний отримати логін і пароль, щоб мати можливість працювати в системі. Для отримання авторизаційних даних потрібно пройти реєстрацію.

На рисунку 3.5 представлена форма реєстрації.

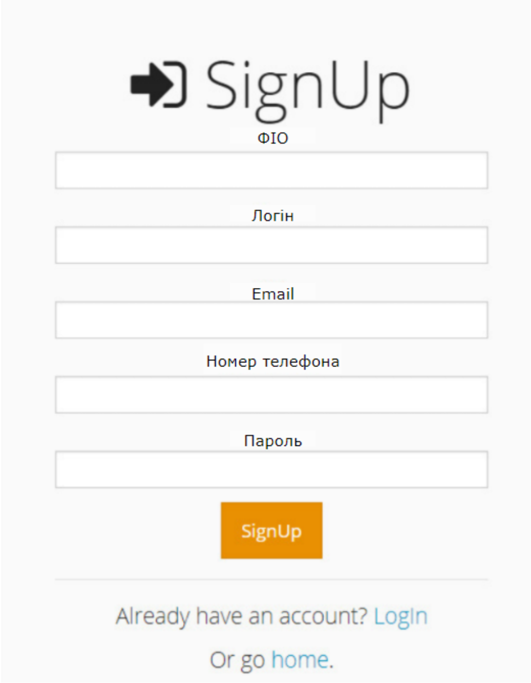

Рисунок 3.5 - форма реєстрації

Після реєстрації нового користувача система перенаправляє на сторінку авторизації, де він може здійснити вхід в систему.

На рисунку 3.6 представлена сторінка авторизації.

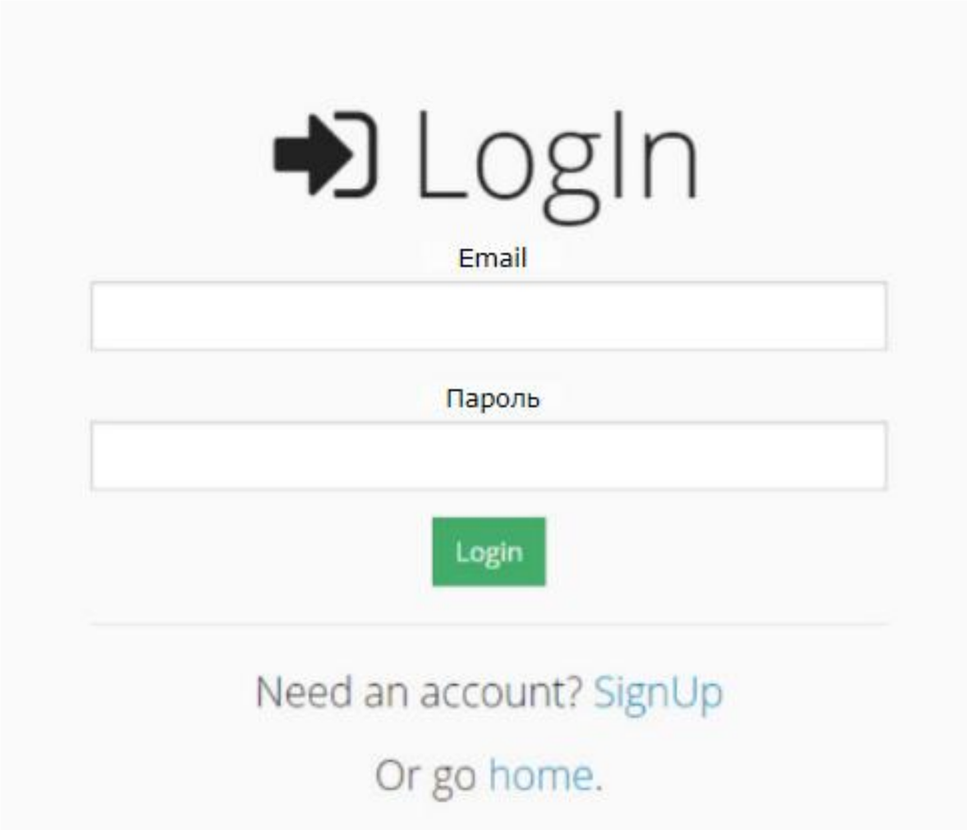

Рисунок 3.6 – Сторінка авторизації

Після входу в систему користувач потрапляє на сторінку профілю, де може переглядати всі висновки про свій стан здоров'я.

На рисунку 3.7 представлена сторінка профілю.

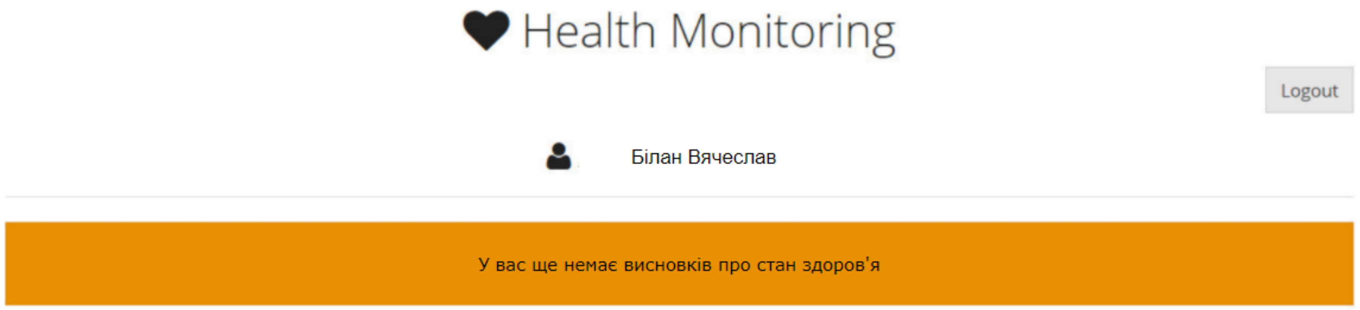

Рисунок 3.7 – Сторінка профілю

На даному етапі профіль порожній і немає ніяких висновків, так як користувач ще не знімав показань обстежень.

Для створення нового висновку Користувачеві необхідно відкрити додатку Chrome і здійснити зняття показань пульсометра.

При запуску плагіна Користувачеві відкриється вікно авторизації.

На рисунку 3.8 представлено вікно авторизації додатку Chrome.

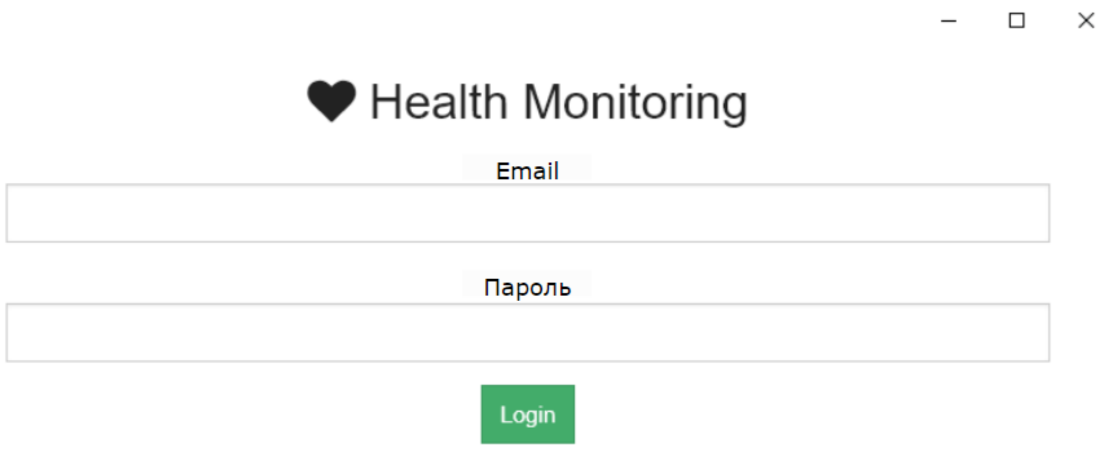

Рисунок 3.8 – Вікно авторизації додатку Chrome

Після авторизації Користувачеві відкриється основне вікно плагіна, якому він може здійснити автопідключення до пульсометру.

На рисунку 3.9 представлено основне вікно плагіна.

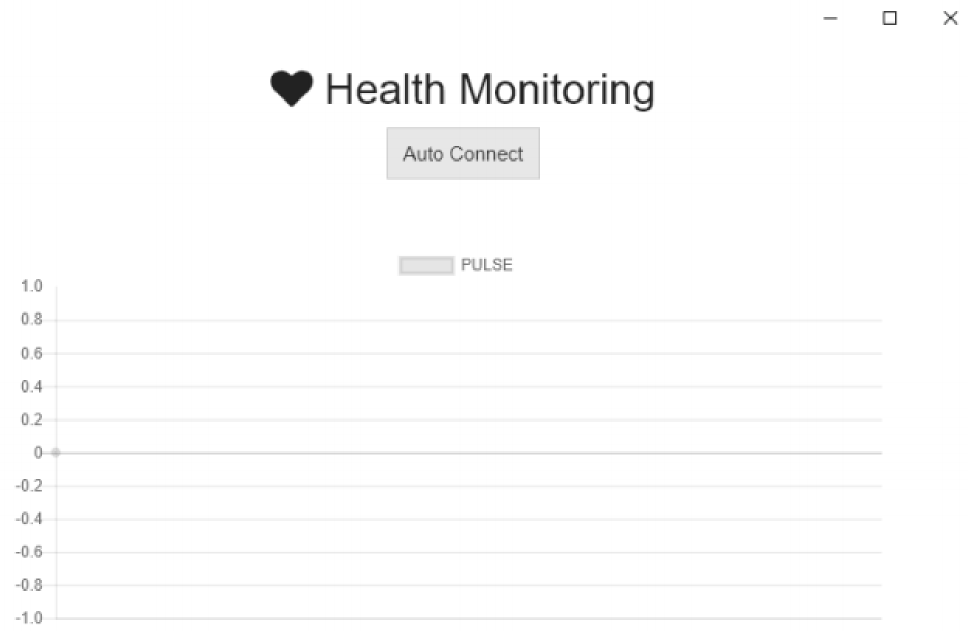

Рисунок 3.9 - Основне вікно додатку Chrome

Після підключення пульсометра користувач може спостерігати пульсограмму в реальному часі.

На рисунку 3.10 представлена пульсограма в реальному часі.

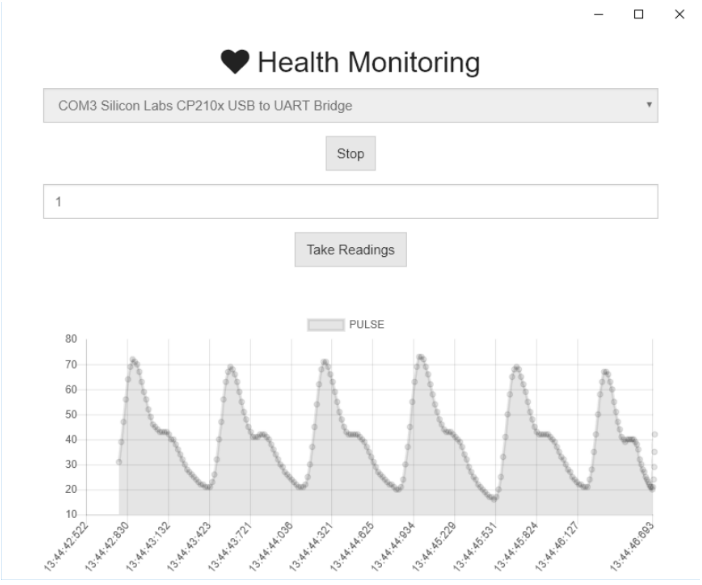

Рисунок 3.10 - Пульсограма в реальному часі

Для зняття показань пульсометра Користувачеві необхідно визначити за який проміжок часу буде проводитися аналіз. В поле введення потрібно ввести обраний інтервал у хвилинах і запустити зняття показань.

На рисунку 3.11 показаний процес зняття показань пульсометра.

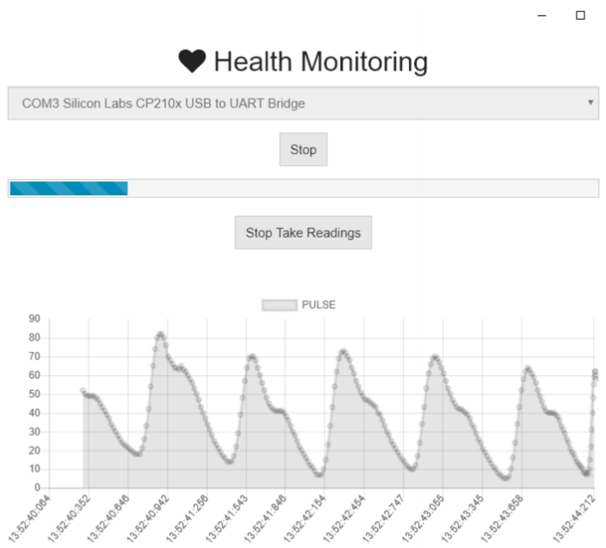

Рисунок 3.11 - Зняття показань пульсометра

Після запуску зняття показань з'являється індикатор, який відображає процес виконання.

При вдалому закінченні зняття показань з'являється повідомлення з інформацією про відправку результатів і можливість повторного зняття.

На рисунку 3.12 показано вікно вдалого зняття показань.

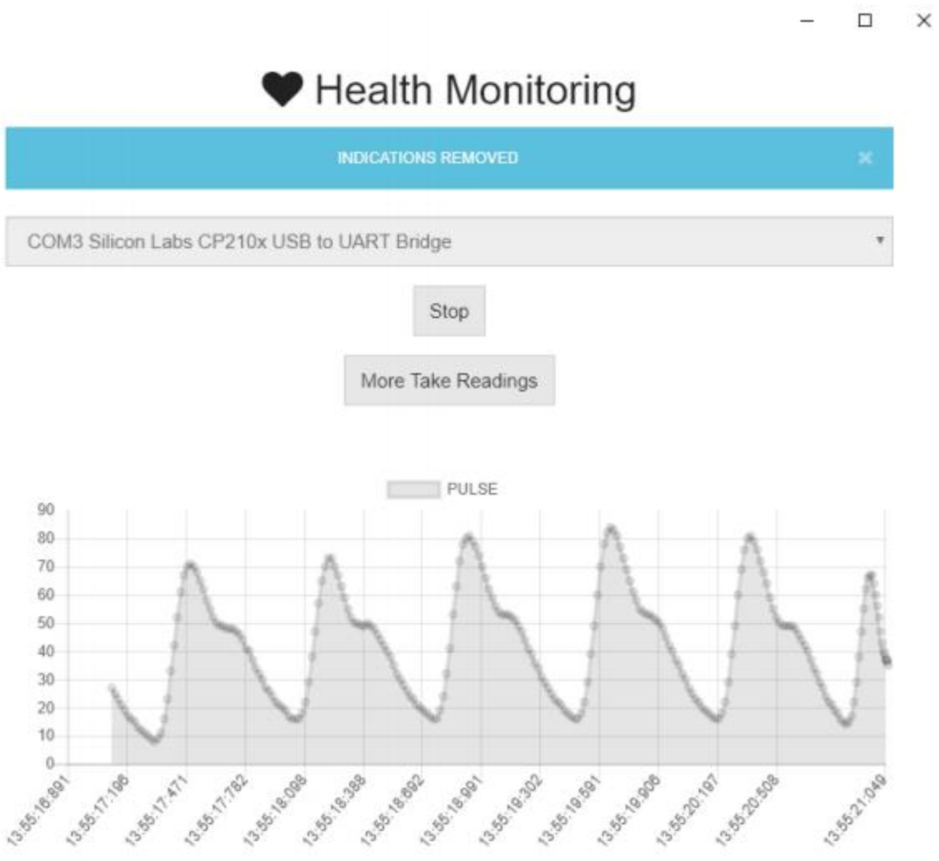

Рисунок 3.12 - Вікно вдалого зняття показань

Після зняття показань пульсометра користувач може зайти на сторінку профілю для ознайомлення з результатами.

На рисунку 3.13 показана сторінка профілю зі списком обстежень і укладення.

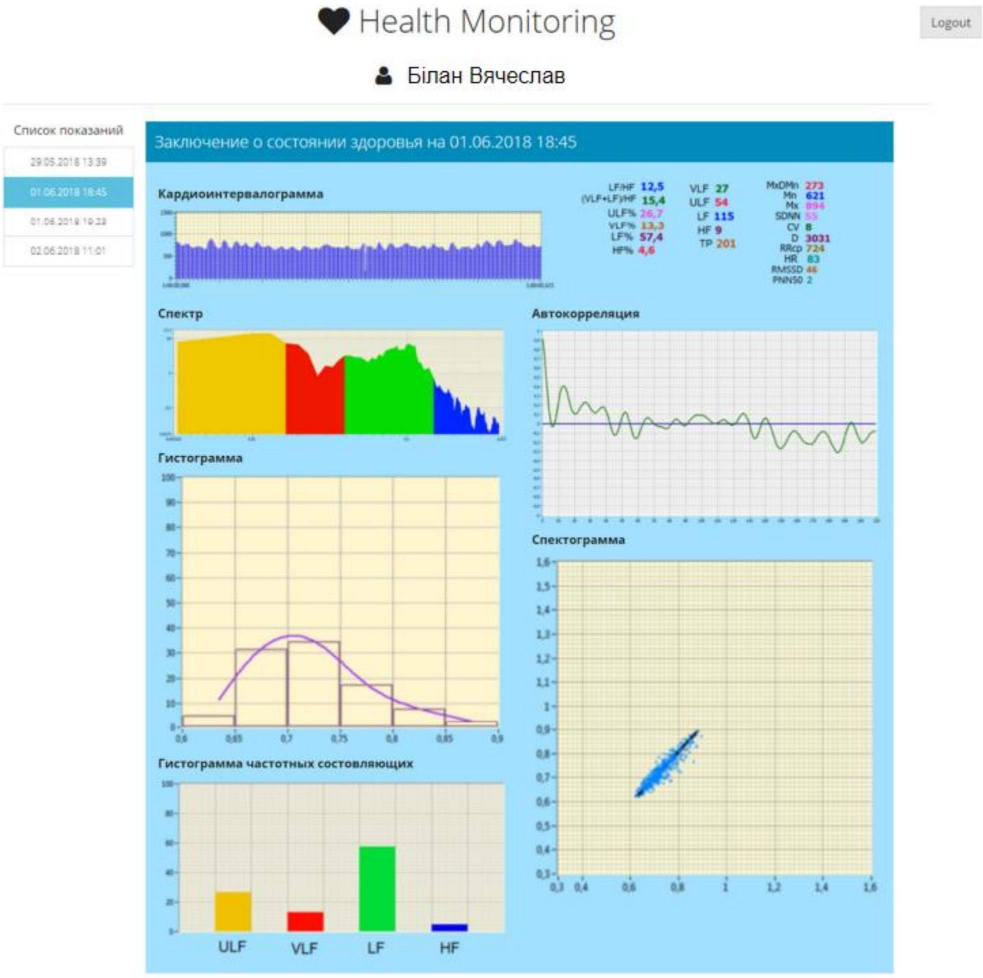

Рисунок 3.13 - Сторінка профілю зі списком обстежень

У заключному розділі слід обґрунтувати ефективність розробки. Це є найважливішим етапом, так як саме даний розділ допомагає відповісти на питання про те, для чого конкретно проводилося дослідження і що конкретно покращився в результаті. Отримана автоматизована інформаційна система найбільше проявила себе як соціально ефективна, так як її основним завданням було дистанційне визначення поточного стану здоров'я.

Соціальна ефективність розробки полягає в тому, що вона сприяє комфортному моніторингу стану здоров'я без зайвих витрат.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи були розглянуті та проаналізовані існуючі моніторингові системи, smart пристрої та додатки для прийому

Виконано розробку інформаційної моделі нової програмної системи, в якій здійснено, рішення проблеми, виявлені на попередньому етапі, розроблена структура бази даних, яка представлена у вигляді инфологической моделі, суті якої відповідають реальним об'єктам предметної області, описані технології розробки серверної частини і клієнтської візуальної оболонки.

Розроблена ієрархічна структура роботи системи, порядок виклику процедур і функцій, принцип роботи діалогу системи і користувача, розділений функціонал системи, що розробляється на модулі, вибрано технологічне забезпечення, що використовується при розробці, наведено тестовий приклад роботи системи, виконано обгрунтування роботи.

У даній роботі були детально описані всі етапи, які необхідно реалізувати на шляху до отримання програмного продукту - автоматизованої системи моніторингу стану здоров'я людини.

ПЕРЕЛІК ПОСИЛАНЬ

1. Барановский А.Л., Немирко А.П. Кардиомониторы. Аппаратура непрерывного контроля ЭКГ. //М.: Радио и связь, 1993. – 247с.

2. Калядин Н.И., Леменков В.А., Коробейников А.В. и другие. Разработка и опыт клинической эксплуатации мониторно-компьютерной системы отделения неотложной реанимации.// Медицинская техника. – 2002. – №1. – С. 36-40.

3. Л. Хейлиг и С. Восс, «Наукометрический анализ литературы по облачным вычислениям», IEEE Transactions on Cloud Computing, vol. 2, вып. 3. С. 266–278, 2014.

4. К. Гай и С. Ли, «На пути к облачным вычислениям: обзор литературы по облачным вычислениям и тенденциям их развития», в Четвертой Международной конференции по мультимедийным информационным сетям и безопасности 2012 г., вып. 146, стр. 142, Нанкин, Китай, ноябрь 2012 г.

5. П. Мелл и Т. Гранс, Национальный институт рекомендаций NIST по определению облачных вычислений. стандартов и технологий, т. 145, Специальная публикация NIST, Гейтерсбург, Мэриленд, США, 2011.

6. Артюшина, Е.А. Разработка web-приложения с использованием архитектуры «клиент-сервер» / Е.А.Артюшина, Е.И.Маркин, К.М.Рябова. // – Международный студенческий научный вестник. – 2016. – №. 3-1. – С. 84-86.

7. Маклаков, С.В, BPwin и Erwin. CASE-средства для разработки

информационных систем / С.В.Маклаков. – Москва: Диалог-МИФИ, 2000. – С. 256.

8. Матюшечев, Л.Ю. Моделирование информационных систем: учебное пособие / Л.Ю.Матюшевич, А.В.Флегонтов // – Санкт-Петербург: Лань, 2018 .– С. 112.

9. Мейер, Б. Основы объектно-ориентированного программирования: учебное пособие / Б.Мейер. – Москва: НОУ "Интуит", 2016. – С. 970.

10. Мюллер, Р.Д. Базы данных и UML. Проектирование / Р.Д. Мюллер. – Москва: ЛОРИ, 2015. – С. 420.

11. Фуфаев, Д.Э. Разработка и эксплуатация автоматизированных информационных систем / Д.Э.Фуфаев, Э.В.Фуфаев. // – Москва: Академия, 2014. – С. 304.

12. Хаббард, Дж. Автоматизированное проектирование баз данных / Дж. Хаббард. – Москва: Мир, 2013. – С. 296.

13. Осипов, А.Л. Информационная система учета и анализа индивидуальных планов: инновационные подходы в решении проблем современного общества / А.Л.Осипов, В.П.Трушина. – Москва: ИНФРА-М, 2017. – С. 192.

14. Закас, Н.С. Ajax для професионалов / Н.С.Закас, Д.Ф.Мак-Пик, Д.Фосетт. – Санкт-Петербург: Символ-Плюс, 2013. – С. 488.

15. Андерсон, Д.Э. Канбан. Альтернативный путь в Agile / Д.Э.Андерсон. – Москва: Манн, 2017. – С. 350.

16. Анохин, П.К. Принципиальные вопросы общей теории функциональных систем. Принципы системной организации функций / П.К.Анохин. – Москва: Наука: 1973. – С.5-61.

17. Баевский, Р.М. К проблеме прогнозирования функционального состояния человека в условиях длительного космического полета / Р.М.Баевский. – Физиол. Журн. СССР. – 1972. – №. 6. – С.819-827.

18. Баевский, Р.М. Кибернетический анализ процессов упраления сердечным ритмом. Актуальные проблемы физиологии и патологии кровообращения / Р.М.Баевский. – Москва: Медицина. – 1976. – С. 161-175.

19. Баевский, Р.М. Математический анализ изменений сердечного ритма при стрессе / О.И. Кириллов, С.З. Клецкин, Р.М. Баевский// – Москва: Наука. – 1984. – С. 220.

20. С. Панді, В. Воорслуйс, С. Ніу, А. Хандокер та Р. Буйя, “Автономне хмарне середовище для розміщення послуг аналізу даних ЕКГ”, Future Generation Computer Systems, vol. 28, ні. 1, с. 147–154, 2012.

21. Ю. Вей, К. Сукумар, Ч. Веккіола, Д. Карунамурті та Р. Буйя, “Хмарна платформа додатків Aneka та її інтеграція з Windows Azur”, у хмарних обчисленнях: методологія, системи та додатки, Л. Ванг , Р. Ранджан, Дж. Чен та Б. Бенаталла, ред., CRC Press, Boca Raton, FL, США, 2011.

ДОДАТОК А

Відомість матеріалів кваліфікаційної роботи

<table>
<tr><td></td><td></td><td>Позначення</td><td>Найменування</td><td>Кількість</td><td>Примітка</td></tr>
<tr><td></td><td>1</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>2</td><td></td><td>Документація</td><td></td><td></td></tr>
<tr><td></td><td>3</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>4</td><td>ІТКІ. ДР 15.01.ДА.ПЗ</td><td>Пояснювальна записка</td><td>1</td><td></td></tr>
<tr><td></td><td>5</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>6</td><td></td><td>Диск CD-R з презентацією</td><td>1</td><td></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td></td><td colspan="5" rowspan="3">ІТКІ.ДП 15.01.ДА.ПЗ</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Зм</td><td>Лист</td><td>№ докум.</td><td>Підпис</td><td>Дата</td></tr>
<tr><td colspan="2">Розроб.</td><td>Білан В.Р.</td><td></td><td></td><td rowspan="5">Матеріали кваліфікаційної роботи</td><td colspan="3">Літ.</td><td>Аркуш</td><td>Аркушів</td></tr>
<tr><td colspan="2"></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>1</td></tr>
<tr><td colspan="2">Керівник</td><td>Гнатушенко</td><td></td><td></td><td colspan="5" rowspan="3">НТУ «Дніпровська політехніка», ФІТ, гр. 123-19м-1</td></tr>
<tr><td colspan="2">Н.контр.</td><td>Цвіркун</td><td></td><td></td></tr>
<tr><td colspan="2">Зав. каф.</td><td>Гнатушенко</td><td></td><td></td></tr>
</table>

ДОДАТОК Б

```
index.js
module.exports = {
	connection: 'mongodb://localhost:27017/heartbeatdb',

	interval: 10000,

	logentries: {
		token: null
	},

	monitor: {
		http: [
			{
				url: 'https://likeastorea.com'
			},
			{
				url: 'https://stage.likeastore.com'
			}
		],

		json: [
			{
				url: 'https://app.likeastore.com/api/monitor',
				response: {
					"app":"app.likeastore.com",
					"env":"production",
					"version":"0.0.52",
					"apiUrl":"/api"
				}
			}
		],

		mongo: [
			{
				connection: 'mongodb://localhost:27017/likeastoredb',
				collections: ['users'],
				query: function (db, callback) {
					db.users.findOne({email:
'alexander.beletsky@gmail.com'}, callback);
				}
			}
		],

		resolve: [
			{
				name: 'google.com'
			}
		],

		ping: [
			{
				ip: '37.139.9.95'
			}
		]
	},

	// notification options
	notify: {
		email: {
```

```
				from: 'heartbeat@likeastore.com',
				to: ['devs@likeastore.com']
			},

			sms: {
				to: ['+3805551211', '+3805551212']
			}
		},

		transport: {
			mandrill: {
				token: 'fake-token'
			},

			twilio: {
				sid: 'fake-sid',
				token: 'fake-token'
			}
		}
	};
```