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

Навчально-науковий
інститут електроенергетики
(інститут)
Факультет інформаційних технологій
(факультет)
Кафедра інформаційних технологій та комп'ютерної інженерії
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня бакалавра

студента **Бабенка Олександра Анатолійовича**
(ПІБ)

академічної групи 123-20-1
(шифр)

спеціальності 123 Комп'ютерна інженерія
(код і назва спеціальності)

за освітньо-професійною програмою 123 Комп'ютерна інженерія
(офіційна назва)

на тему «IoT система медичного обслуговування хворих на цукровий діабет»

(назва за наказом ректора)

Керівники	**Прізвище, ініціали**	**Оцінка за шкалою**		**Підпис**
		рейтинговою	**інституційною**	
кваліфікаційної роботи	доц. Бешта Д.О.			
спеціальної частини	ас. Панферова Я.В.			
розділів:				
розробка апаратної частини	доц. Ткаченко С.М.			
розробка корпоративної мережі	ас. Бешта Л.В.			

Рецензент				

Нормоконтролер	проф. Цвіркун Л.І.			

Дніпро
2024

ЗАТВЕРДЖЕНО:

завідувач кафедри
інформаційних технологій
та комп'ютерної інженерії
(повна назва)

_____ Гнатушенко В.В.
(підпис) (прізвище, ініціали)
« » _____ 2024 року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня бакалавр

студента Бабенка О.А. **академічної групи** 123-20-1
прізвище та ініціали) (шифр)

спеціальності 123 Комп'ютерна інженерія

за освітньо-професійною програмою 123 Комп'ютерна інженерія
офіційна назва)

на тему «IoT система медичного обслуговування хворих на цукровий діабет»

затверджену наказом ректора НТУ «Дніпровська політехніка» від 29.04.2024 № 375-с

Розділ	**Зміст**	**Термін виконання**
Стан питання та постановка завдання	На основі матеріалів виробничих практик, інших науково-технічних джерел розглянути призначення та завдання IoT системи медичного обслуговування хворих на цукровий діабет	10.05.2024
Розробка апаратної частини	Розробити вимоги до функцій, виконуваними системою, розробити структурну схему та специфікацію обладнання	20.05.2024
Розробка корпоративної мереж	Побудувати в Packet Tracer модель корпоративної мережі компанії, виконати налаштування та перевірку роботи системи	10.06.2024
Розробка компонента системи	Розробити IoT систему медичного обслуговування хворих на цукровий діабет	20.06.2024

Завдання видано _____ доц. Бешта Д.О.
(підпис керівника) (прізвище, ініціали)

Дата видачі 01.05.2024

Дата подання до екзаменаційної комісії 02.07.2024

Прийнято до виконання _____ Бабенко О.А
(підпис студента) (прізвище, ініціали

РЕФЕРАТ

Пояснювальна записка: 89 с., 54 рис., 9 табл., 1 дод., 10 джерел.

КОМП'ЮТЕРНА IoT СИСТЕМА, ІНТЕРНЕТ РЕЧЕЙ, ЦУКРОВИЙ ДІАБЕТ, МАРШРУТИЗАТОР, КОМУТАТОР, CISCO, PACKET TRACER, NAT, VPN, DHCP, VLAN.

Об'єкт розробки – IoT система медичного догляду за пацієнтами з цукровим діабетом з реалізацією побудови та налаштування корпоративної мережі.

Мета роботи – створення IoT системи для потенційного медичного закладу, що піклується про здоров'я пацієнтів хворих на ЦД.

Було розроблено корпоративну мережу, що може гнучко змінювати свій зовнішній вигляд, кількість обслуговуючих пристроїв і набір функцій шляхом додавання нових пристроїв та налаштування їх під потреби організації.

Дана IoT система дозволяє здійснювати як технічну, так і програмну модернізацію системи. Організаційна структура системи складається з 5 відділів:

– технічна підтримка та обслуговування;

– мобільна група реагування;

– ліагностичне відділення;

– адміністрація та управління;

– навчальний центр.

Розроблена корпоративна мережа виконана відповідно до завдання на кваліфікаційну роботу бакалавра, з урахуванням усіх поставлених вимог.

Розроблена схема мережі виконана та перевірена за допомогою програми Cisco Packet Tracer. Результати перевірки у вигляді таблиць, графіків описані і наводяться у пояснювальній записці та додатках.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, ОДИНИЦЬ І ТЕРМІНІВ

БД – база даних

ІТ – інформаційні технології

ПЗ – програмне забезпечення

СЦ – сервісний центр

ТЗ – технічне завдання

ШІ – штучний інтелект

IoT – Інтернет Речей

МОХЦД – медичне обслуговування хворих на цукровий діабет

PT – Packet Tracer

МГР – мобільна група реагування

ВСТУП

У сучасному світі IoT стає все більш важливою складовою технологічного прогресу, стрімко розвиваючись, набираючи популярності та забезпечуючи підключення різних роду пристроїв і систем для автоматизації задля полегшення різних аспектів життя. У сфері медичних технологій IoT відіграє особливо значущу роль, він дозволяє вдосконалити процес нагляду за пацієнтами та забезпечити їм ефективне і точне медичне обслуговування, а також контроль за їх станом здоров’я.

Так як цукровий діабет є однією з найбільш поширених у світі хронічних захворювань, що вимагає постійного моніторингу та управління рівнем цукру в крові, то розробка IoT системи для медичного обслуговування пацієнтів хворих на цукровий діабет є актуальним завданням, спрямованим на поліпшення якості життя таких пацієнтів та оптимізацію процесу їхнього лікування.

У цій кваліфікаційній роботі пропонується розробка галузі що складається з IoT системи, у якості моделяції використовуємо програмне забезпечення Packet Tracer. Запропоноване рішення спрямоване на моніторинг та управління показниками цукру в крові пацієнтів з цукровим діабетом. Детально буде розглянуто як апаратна частина так і мережева, основні компоненти та їх взаємодію.

Це представлене рішення спрямоване на поєднання технологічних можливостей з потребами сучасної медицини, маючи на меті покращення якості надання медичних послуг та забезпечення пацієнтів з цукровим діабетом надійними та зручними інструментами для керування та піклування про своє здоров'я.

1 СТАН ПИТАННЯ І ПОСТАНОВКА ЗАВДАННЯ

1.1 Стисла характеристика галузі піклування про пацієнтів хворих на цукровий діабет

Галузь піклування про пацієнтів хворих на цукровий діабет зосереджується на інноваційних технологіях, спрямованих на покращення якості життя та здоров'я пацієнтів з цукровим діабетом. За останні роки спостерігається стрімкий розвиток сучасних технологій, зокрема Internet of Things (IoT), які забезпечують медичні заклади та пацієнтів з інноваційними можливостями для моніторингу та управління за станом здоров’я хворих.

Основні аспекти галузі включають в себе використання сенсорів, датчиків та зв'язаної з ними мережної інфраструктури для безперервного збору та аналізу даних про рівень глюкози в крові, фізичну активність, харчування та інші важливі параметри здоров'я, щоб вчасно можна було надати допомогу, або передбачити зміни у організмі. Інтеграція цих даних в медичні системи дозволяє не лише вчасно виявляти та прогнозувати погіршення стану пацієнта, але й забезпечує можливість автоматичної реакції на виявлені аномалії, наприклад, надання рекомендацій з дієтою чи призначення необхідного лікування.

Така інноваційна система медичного піклування дозволяє пацієнтам з цукровим діабетом зберігати більший контроль над своїм станом здоров'я, зменшуючи ризик розвитку ускладнень та підвищуючи їх якість життя. Крім того, вона сприяє зниженню навантаження на медичних працівників, дозволяючи їм зосередитися на наданні якісної медичної допомоги, в той час як система автоматично виконує моніторинг та аналіз великої кількості даних.

1.2 Характеристика закладу з галузі піклування про людей хворих на цукровий діабет із залученням технологій IoT

Характеристика типічного закладу в галузі піклування про людей, хворих на цукровий діабет із залученням технологій IoT, повинна показувати наявність інноваційного підходу до надання медичних послуг та підтримки пацієнтів. Цей

заклад, що спеціалізується на розробці та впровадженні IoT-рішень у сфері охорони здоров'я, повинен активно використовувати передові технології для поліпшення якості життя та піклування про пацієнтів з цукровим діабетом.

Заклади повинні розробляти та впроваджувати інтегровані системи моніторингу та управління за станом здоров'я пацієнтів, які забезпечують постійний контроль за рівнем глюкози в крові, фізичною активністю та іншими важливими показниками. Ці рішення базуються на сучасних сенсорах, забезпечують зручний і надійний збір даних, а також використовують аналітичні алгоритми для швидкого виявлення та аналізу аномалій.

Крім того, вони мають пропонувати інтелектуальні рішення для підтримки пацієнтів у керуванні їхнім хворобливим станом, забезпечуючи автоматичну реакцію на виявлені відхилення та надаючи рекомендації з дієтою та лікуванням, враховуючи сучасні тенденції, забезпечити це можливо завдяки залученню ШІ.

Також заклад має мати широкий досвід у розробці та впровадженні IoT-систем у медичній галузі та готовий до співпраці з клієнтами для створення індивідуальних рішень, що відповідають їхнім потребам та вимогам.

1.3 Принципи, технічні способи та математичні методи інформаційного забезпечення галузі

Принципи, технічні способи та математичні методи інформаційного забезпечення галузі піклування про людей, хворих на цукровий діабет, відіграють важливу роль у забезпеченні ефективного моніторингу та управління за їхнім станом здоров'я. Для вирішення цієї проблеми існують різноманітні варіанти рішень, які використовуються в медичній практиці, проте, звісно, кожен з цих варіантів має свої переваги та недоліки які ми розглянемо далі.

Один з найбільш поширених та ефективних варіантів рішення полягає у використанні носимих пристроїв, таких як глюкометри та фітнес-трекери (Smart-годинники), які забезпечують постійний моніторинг рівня глюкози в крові та фізичної активності. Ці пристрої часто синхронізуються з мобільними додатками, що дозволяють пацієнтам вести історію (журнал) своїх показників та

отримувати рекомендації щодо дієти та лікування. Однак ці рішення можуть бути обмежені в точності та надійності даних, а також не враховувати індивідуальні особливості захворювання пацієнтів, тому є сенс у знаходженні інших рішень, або допрацювань вже наявних.

Інший підхід полягає у використанні імплантованих (влаштованих) пристроїв, які автоматично моніторять рівень глюкози в крові та інші важливі показники, такі як частота дихання. Ці пристрої надають більш точні та надійні дані, але вони можуть бути дорогими та вимагати хірургічного втручання для встановлення, а це може не завжди влаштовувати потенційного пацієнта.

Проте, незважаючи на ці два запропонованих варіантів рішень, деякі проблеми ще залишаються невирішеними. Наприклад, існує потреба у розвитку алгоритмів щодо аналізу даних, які б забезпечували точні прогнози стану пацієнта та рекомендації з управління хворобою. Крім того, було б важливо забезпечити інтеграцію між різними медичними системами та платформами, щоб забезпечити ефективний обмін даними між лікарями та пацієнтами, можливо створення одного єдиного ресурсу, що дозволяв би швидко та надійно вирішувати поставлені потреби.

Також необхідно вдосконалювати засоби забезпечення конфіденційності та захисту особистих даних пацієнтів, оскільки це є критично важливим аспектом в галузі медичної інформатики, а це питання вже відноситься і до сфери кіберзахисту.

1.4 Організаційна структура середньостатистичного закладу медичного піклування

Організаційна структура закладу медичного піклування про людей (рис.1.1), хворих на цукровий діабет із залученням технологій IoT, включало б різні рівні та підрозділи, що співпрацювали б для забезпечення ефективної допомоги пацієнтам та впровадження інноваційних технологій, щоб покращити сервіс.

а) Адміністрація та управління. Цей рівень включає керівництво закладу, яке визначає стратегічні цілі та політику впровадження нових технологій або технологій IoT. Адміністративний персонал також відповідає за фінансове управління, а також за взаємодію з партнерами та постачальниками технологічних рішень.

б) Відділ інформаційних технологій (IT). Цей відділ відповідає за розробку, впровадження та підтримку технічних рішень IoT в медичному закладі, наприклад написання коду для IoT-рішень. IT-спеціалісти забезпечують інтеграцію та оптимізацію мережевої інфраструктури, а також розвиток програмного забезпечення для збору, аналізу та візуалізації даних.

в) Медичний персонал. Лікарі, медичні сестри та інший медичний персонал, що використовують технології IoT для моніторингу пацієнтів, надання телемедичної консультації та надання рекомендацій щодо дієти та лікування.

г) Мобільна група реагування (МГР). Це спеціальний підрозділ медичних фахівців, що включає медичний персонал, водія, і IT-спеціаліста зі впроваджених технологій якими забезпечені пацієнти. Ця група предназначена для швидкого реагування у разі погіршення стану пацієнта для надання йому невідкладної медичної допомоги.

д) Діагностичне відділення. Цей підрозділ відповідає за збирання та аналіз біомедичних даних, зокрема рівня глюкози в крові, які передаються через сенсори та медичні пристрої IoT.

е) Технічна підтримка та обслуговування. Цей відділ забезпечує технічну підтримку для усіх систем та пристроїв IoT, так звані «гарячі лінії», а також відповідає за їхнє обслуговування та ремонт у випадку несправностей (СЦ).

ж) Навчальний центр. Для успішного впровадження технологій IoT у медичному закладі важливо навчати медичний персонал користуватися новими системами та пристроями. Навчальний центр забезпечує навчання та підтримку персоналу.

Ця організаційна структура дозволяє забезпечити ефективне впровадження та використання технологій IoT для покращення медичного піклування за

пацієнтами хворих на цукровий діабет. Кожен відділ взаємодіє з іншими, щоб забезпечити гармонійну та ефективну роботу всього медичного закладу.

Рисунок 1.1 – Схема організаційної структури медичного закладу

1.5 Топологічне розміщення структурних підрозділів

Площа, що повинна бути відведена для закладу у цій сфері, повинна бути не менеше ніж 2500 метрів квадратних, будівля центру піклування про здоров’я має принаймні один поверх де розміщені усі структурні підрозділи, такі як діагностичне відділення, навчальний центр, технічна підтримка, адміністрація, медичні працівники та зала очікування.

Біля самої будівлі безпосередньо повинна знаходитись парковка для машин співробітників, пацієнтів, а також спеціальне парко-місце для декількох мобільних груп швидкого реагування.

Наявність зелених насаджень навкруги бажано, але не обов’язково. Щодо вигляду самої будівлі: вона повинна бути добре освітленою, з великою кількістю вікон, щоб до приміщень потрапляло багато сонячного світла.

Щодо підрозділів, про які було згадано раніше.

а) Кімната дігностичного відділення має на балансі певну кількість персональних комп’ютерів на яких відбувається аналіз та обробка даних, що у подальшому будуть передані на комп’ютери-сервери, де будуть надійно зберігатись.

б) Кімната з технічної підтримки та обслуговування повинна налічувати певну кількість персональних комп’ютерів та відповідного персоналу що будуть надавати консультаційні послуги пацієнтам онлайн/офлайн, а також обслуговувати медичну техніку, таку як смарт-годинники, глюкометри, ПК, тощо. Також за цим відділом закріплене складське приміщення, де розміщується резервна техніка у вигляді медичного обладнання та мережевого обладнання.

в) Кімната адміністрації – це відносно мале приміщення, де розміщені важливі документи, що зберігаються в сейфі бухгалтерії. Кімната обладнана малою кількістю персональних комп’ютерів.

г) Кімната навчального центру, це місце, де розташована невелика кількість персональних комп’ютерів та іншого медичного обладнання, що призначене для демонстраційного показу та навчання медичного персоналу роботи з ними. Також тут відбувається кампанія по набору персоналу для центру піклування про здоров’я пацієнтів хворих на цукровий діабет.

д) Кімната медичних працівників (так звана ординаторська), де розташований один-два персональних комп’ютери. У цій кімнаті як правило знаходиться медичний персонал у часи перерви або чергування.

е) І найбільша кімната у цілому центрі – це зала очікування, зала, де розташовані місця для сидіння, різні табло з інформацією, термінали, а також пост охорони, що знаходиться на вході у приміщення. Охорона обладнана персональним комп’ютером та системою відеоспостереження, що встановлена по всьому майданчику центру, як всередині, так і ззовні на парковці.

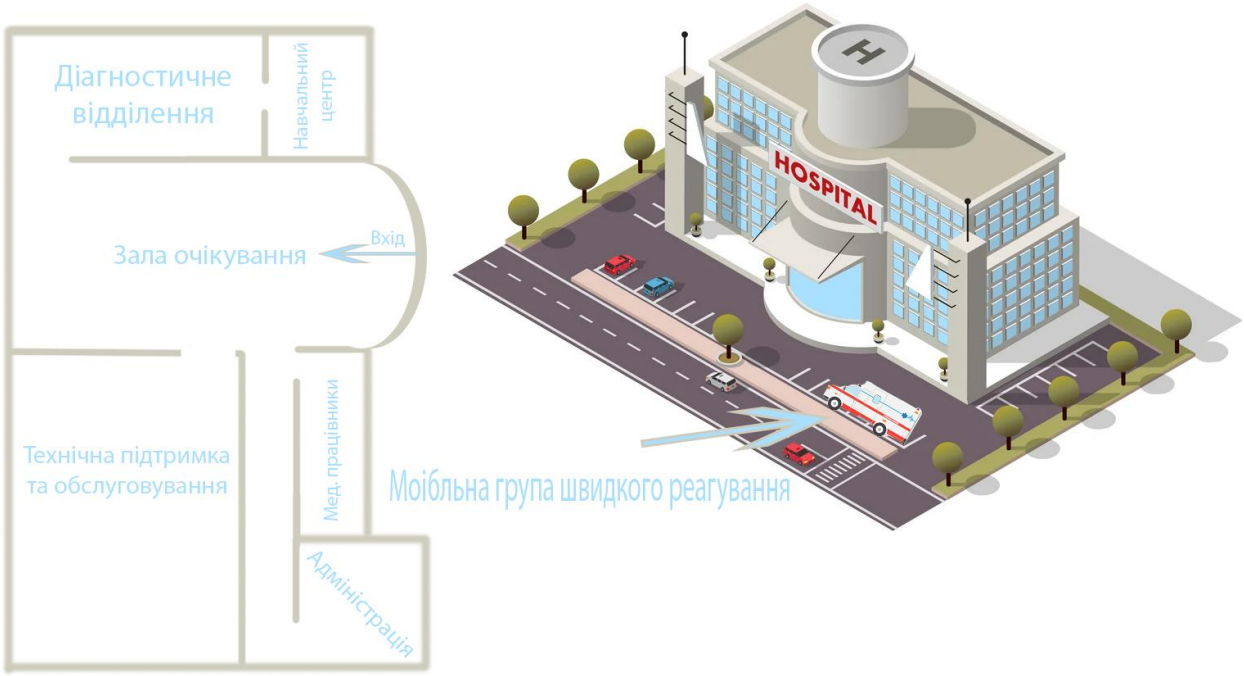

Рисунок 1.2 – Топологічна схема розміщення підрозділів у лікарні

1.6 Визначення можливих напрямків рішення поставлених задач

Наступні напрямки можна комбінувати для створення комплексної IoT системи, що зможе забезпечити ефективне медичне обслуговування людей з цукровим діабетом.

а) Носимі пристрої та датчики, а саме розробка носимих пристроїв, таких як розумні годинники або сенсорні пластирі, які можуть вимірювати рівень глюкози в крові, пульс, рівень активності та інші важливі параметри. Ці дані можуть передаватися через IoT на медичний сервер закладу для моніторингу.

б) Інтернет речей для медичних пристроїв, конкретно розробка медичних пристроїв, які можуть бути підключені до Інтернету і надсилати дані про стан хворого, наприклад, інсулінові помпи, які можуть регулювати рівень інсуліну залежно від показників глюкози в крові.

в) Системи моніторингу та аналізу даних, мається на увазі розробка спеціалізованих програмних засобів для моніторингу та аналізу даних, що були зібрані з пристроїв IoT. Ці системи можуть використовувати алгоритми штучного

інтелекту (ШІ) для передбачення ризиків або для надання рекомендацій щодо лікування.

г) Платформи з обміну даними між пацієнтами, лікарями та іншими медичними працівниками. Це дозволить підвищити співпрацю та координацію у догляді за хворими на цукровий діабет.

д) Розробка механізмів захисту персональних медичних даних, які передаються через IoT, для забезпечення конфіденційності та захисту від несанкціонованого доступу.

1.7 Обґрунтування вибраного напрямку інженерного рішення

Із стрімким розвитком Інтернету Речей (IoT) все більше спеціалістів в галузі інформаційних технологій звертають увагу на необхідність ефективного моделювання та симуляції мережевих систем. Тому для моделювання IoT системи медичного обслуговування хворих на цукровий діабет та мережі медичних лікарень було прийнято рішення використовувати програму для моделювання IoT-речей: Cisco Packet Tracer.

По-перше, Packet Tracer (PT) дозволяє створювати складні мережеві топології, що імітують реальні IoT-системи. Це дає можливість аналізувати взаємодію різноманітних пристроїв, перевіряти ефективність протоколів та алгоритмів, а також оптимізувати продуктивність мережі ще на етапі проектування. Таким чином, використання Packet Tracer істотно скорочує витрати на розгортання та тестування IoT-систем.

По-друге, Packet Tracer має широкий спектр інструментів для налаштування, моніторингу та діагностики мережевих пристроїв. Це дозволяє спеціалістам ретельно досліджувати особливості функціонування IoT-компонентів, виявляти та усувати можливі проблеми ще до впровадження реальної системи.

По-третє, Packet Tracer є безкоштовним програмним забезпеченням для студентів Мережної академії Cisco, що робить його доступним для широкого кола користувачів.

Підсумовуючи, Packet Tracer є ефективним інструментом для моделювання Інтернету Речей, що дозволяє оптимізувати процеси проектування, налаштування та експлуатації IoT-систем. Його використання сприяє зниженню витрат, підвищенню надійності та продуктивності IoT-рішень, а також популяризації цієї технології серед спеціалістів у сфері інформаційних технологій.

Для моделювання мережі лікарень було вирішено обрати топологію на рисунку 1.3. Для цього проекту потрібно створити п’ять локальних мереж. Кожна мережа буде моделювати окрему мережу медичного закладу в різних районах або містах.

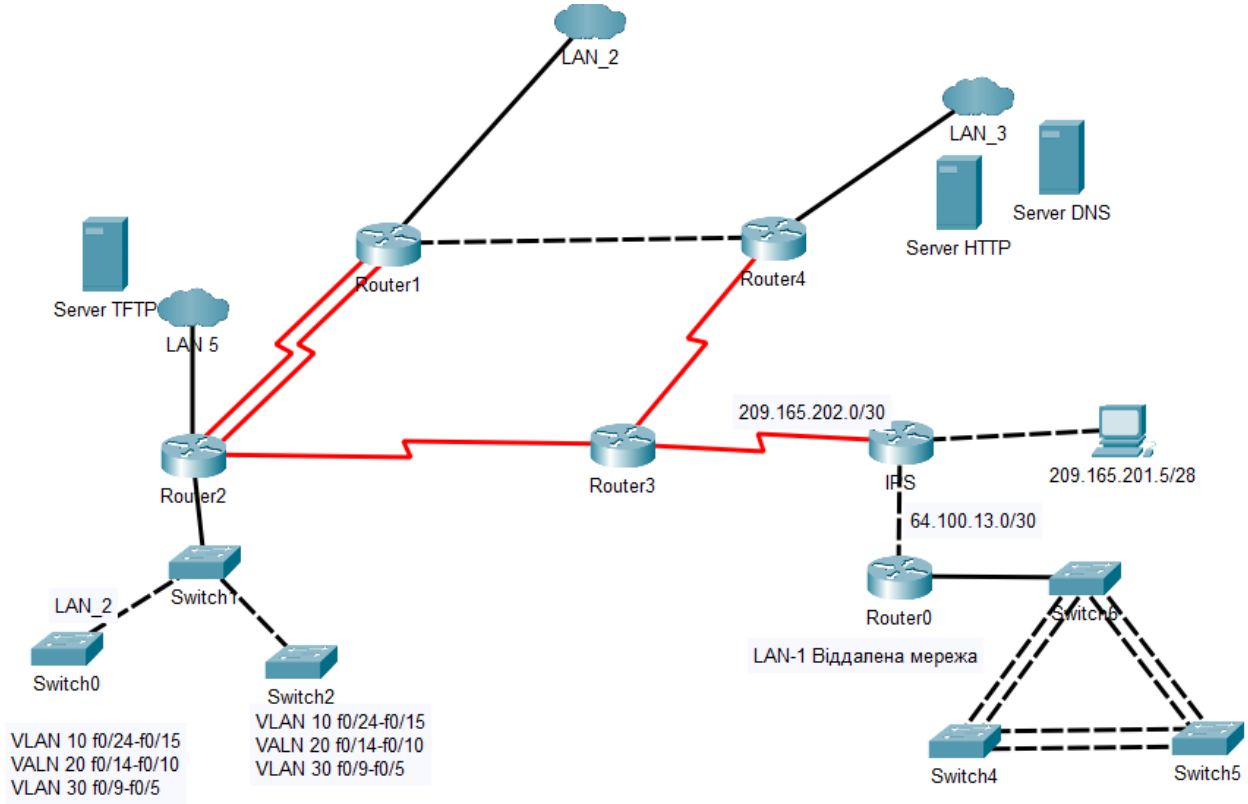

Рисунок 1.3 – Загальна топологія мереж лікарень

1.8 Постановка завдання

Завданням кваліфікаційної роботи є створення опису галузі, розробки методів та структурних організацій для допомоги людям хворих на цукровий діабет.

Для вирішення цього завдання необхідно виконати наступні пункти:

– ознайомитись із організаційною структурою типового медичного закладу;

– розглянути галузь допомоги людям хворим на діабет;

– зробити аналіз галузі, визначити наявні рішення проблеми з використанням IoT-технологій;

– розробити схему «Схема гео-розміщення компонентів інформаційної системи»;

– розробити схему «Схема організаційної структури підприємства»;

– розробити схему «Структурна схема комплексу технічних засобів інформаційної системи»;

– формулювання технічних вимог до комп'ютерної системи;

– вказати технічні способи та математичні методи інформаційного забезпечення галузі;

– виконати перелік технічних вимог щодо комп'ютерної системи;

– виконати перелік підсистем;

– сформулювати технічні вимоги до мережі;

– зробити підбір відповідного мережевого обладнання (комутатори, маршрутизатори, тощо);

– розробити специфікацію апаратних засобів;

– побудувати модель мережі медичних закладів в PT;

– налаштувати мережне обладнання;

– розробити IoT систему мобільного медичного обслуговування хворих на цукровий діабет;

– розробити в РТ IoT-датчики для постійного моніторингу рівня глюкози пацієнта;

– розробити в РТ IoT фітнес-трекер для моніторингу рівня фізичного навантаження та дихання пацієнта;

– обрати метод передачі даних до медичного закладу;

– розробити платформу для моніторингу та аналізу даних, що були зібрані з пристроїв IoT;

– розробити платформу з обміну даними між пацієнтами, лікарями та іншими медичними працівниками;

Результатом виконання кваліфікаційної роботи є створення опису структури, підрозділа, цілої системи, що займається піклуванням про здоров’я пацієнтів, що мають захворювання цукрового діабету. Опис структури має містити у собі структурну схему системи, схему гео-розміщення системи та схему організаційної структури, а також заповнене ТЗ.

2 РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ КОМП'ЮТЕРНОЇ СИСТЕМИ

2.1 Технічні вимоги до комп'ютерної системи

2.1.1 Вимоги до системи в цілому

2.1.1.1 Вимоги до структури і функціонуванню системи

IoT-система медичного обслуговування хворих на цукровий діабет (МОХЦД) створена для надання допомоги людям із захворюванням цукрового діабету, моніторингу їхнього стану здоров'я та надання рекомендацій. IoT-систему МОХЦД, повинна забезпечувати:

– моніторинг стану здоров'я пацієнта за допомогою імплантованого IoT-глюкометра;

– надання інформації пацієнту пов'язаної зі зміною стану його здоров'я та переліку рекомендованих дій у разі його погіршення;

– підтримку прямого спілкування між пацієнтом та лікарем;

IoT- система МОХЦД повинна представлятися комплексом технічних і апаратних засобів для досягнення роботи медичного персоналу, а також безпеки здоров'я пацієнта.

2.1.1.1.1 Перелік підсистем, їхнє призначення й основні характеристики, вимоги до числа рівнів ієрархії та ступені централізації

Система Інтернету речей для медичного обслуговування хворих на цукровий діабет складається з підсистем, кожна з яких має свої специфічні призначення й характеристики:

– підсистема моніторингу стану здоров'я пацієнта;

– підсистема обробки та зберігання даних;

– підсистема телемедицини;

Підсистема моніторингу стану здоров'я пацієнта предназначене для постійного контролю рівня глюкози в крові, а також відстеження інших важливих показників таких як артеріальний тиск, частота серцевих скорочень та рівень фізичної активності. Вона включає в себе використання носимих сенсорів та

пристроїв (глюкометри, смарт-годинники, браслети тощо), здійснює автоматичний збір даних і передає їх лікарям у режимі реального часу. Також присутня можливість інтеграції з мобільними додатками для користувачів такими як Helsi та Лікарі 24/7.

Підсистема обробки та зберігання даних має на меті зберігання великих обсягів даних про пацієнтів (дані з датчиків), а також аналіз і обробку даних для виявлення тенденцій та аномалій у стані здоров'я пацієнта. Використовуючи власні IoT-сервери системи забезпечуючи надійність зберігання та конфіденційності даних.

Підсистема телемедицини має на меті забезпечення дистанційного спостереження за станом здоров'я пацієнта та консультацій з лікарем, а також надання швидкої медичної допомоги в екстрених ситуаціях.

Наявність цих підсистем утворюють комплексну IoT-систему, що допомагає управляти лікуванням і моніторингом стану хворих на цукровий діабет, забезпечуючи при цьому високу якість життя пацієнтів.

2.1.1.1.2 Вимоги до способів і засобів зв'язку між компонентами комп'ютерної системи

Функціональний блок, що відповідає за підсистему моніторингу стану здоров'я пацієнта та передачі даних на сервер повинен здійснювати передачу даних через пристрій Home Gateway, що є ядром IoT системи у будинку пацієнта, той пристрій, у свою чергу, повинен бути під'єднаний до мережі Internet використовуючи оптоволоконне дротове підключення.

Комп'ютерні станції, планшети, сервери та інше медичне обладнання також повинно бути під'єднано до мережі Internet за допомогою як дротового, так і бездротового підключення в залежності від виду девайсу.

Мобільна група реагування (МГР), що входить до підсистеми телемедицини, повинна бути забезпечена SIM-картами з підтримкою 4G технології та тарифом, що включає в себе безлімітний доступ до Internet, який

буде оплачуватись адміністрацією, а також один стаціонарний комп’ютер, на який будуть приходити сповіщення про виклики та для відправки звітності щодо виконаних робіт.

Функціональний блок, що відповідає за медичне забезпечення, має бути під’єднано до мережі Internet з використанням високошвидкісного оптоволоконного дротового підключення.

2.1.1.1.3 Вимоги до характеристик взаємозв'язків створюваної системи із суміжними системами

IoT система МОХ на ЦД повинна мати взаємозв’язок із суміжними системами за допомогою мережевого інтерфейсу Ethernet. Враховуючи, що IoT система містить такі девайси як імплантований глюкометр або смарт-годинник, що використовують технологію зв'язку Wi-Fi для створення бездротового підключення до Home Gateway, то система буде під’єднана до суміжної системи, що являє собою корпоративну мережу лікарні.

2.1.1.1.4 Вимоги до режимів функціонування системи

а) Автоматичний режим – основний режим функціонування системи:

– система повинна забезпечувати постійний моніторинг рівня глюкози в крові в режимі реального часу для 5000 пацієнтів одночасно:

– повідомлення про критичні значення рівня глюкози повинні надходити негайно (не пізніше ніж через 1 хвилину після вимірювання);

– інтерфейс користувача (додаток для моніторингу стану здоров’я пацієнта) повинен бути доступний 24/7 з часом відповіді не більше 3 секунд.

б) Відлагоджувальний режим – режим для виявлення та усунення помилок у системі:

– основні функції моніторингу рівня глюкози та сповіщення про критичні значення повинні залишатися активними;

– всі інші функції можуть бути обмежені або недоступні;

– користувачі повинні бути сповіщені про відлагоджувальний режим та приблизний час його завершення;

– система повинна забезпечувати збереження всіх даних з датчиків, зібраних у цьому режимі, для подальшої синхронізації після відновлення робочого режиму;

– система повинна дозволяти введення імітованих даних для виявлення помилок;

– логи повинні містити детальну інформацію про всі операції, що проводяться, з позначками часу;

– доступ до відлагоджувального режиму повинен мати лише відділ технічної підтримки та адміністрація;

– система повинна забезпечувати можливість відкату змін, внесених у цьому режимі, протягом 1 години після їх застосування.

Показники часу:

– час реагування на критичні показники глюкози: до 1 хвилини;

– час відгуку інтерфейсу користувача в робочому режимі: від 2 до 3 секунд;

– час доступності тестових результатів: протягом 24 годин після завершення тестування;

– час відкату змін у відлагоджувальному режимі: до 1 години;

– оновлення інформації користувачів про відлагоджувальний режим: негайно після входу в відлагоджувальний режим.

2.1.1.1.5 Вимоги до діагностування системи

Відповідно до твердження, що система є комплексною, а це значить, що скаладається вона з різних рішень та компонентів, діагностика буде багаторівнева, як фізична (перевірка каналів зв’язку, кабелів, мережевого обладнання тощо) так і програмна, що включатиме в себе перевірки на працездатність системи, відсутність помилок у роботі пристроїв тощо.

Для діагностики неполадок у системі можуть бути використані збережені налаштування пристроїв (резервні копії робочої системи).

Періодичність діганостики має складати не рідше ніж раз на рік.

2.1.1.1.6 Перспективи розвитку, модернізації системи

Модернізація системи передбачає її інтеграцію у модернізовану КС лікарні, яка матиме наступні технічні характеристики:

Мережа складатиметься з 5 підмереж – підрозділів, що повинні містити 321 хост відповідно до завдання замовника, а сама мережа повинна мати наступні налаштування: протоколу динамічного розподілу IP-адрес DHCP, механізм зміни мережевої адреси NAT, створення віртуального захищеного підключення VPN, протколу динамічної маршрутизації OSPF, віртуальних локальних мереж VLAN та застосування методу AAA.

Мережа лікарні має топологію відповідно до завдання на рис.1.3.

Під час розробки адресації підмережі необхідно враховувати наступні вимоги:

– корпоративна мережа повинна складатися з 5 підмереж MED1-MED5, з конкретною кількістю вузлів в кожній з них: 39, 107, 24, 92 та 59 відповідно;

– блок адрес для виділення підмереж повинен бути 10.24.8.0/22;

– для каналів між маршрутизаторами застосувати блок адрес 10.0.13.0/24

– середня інтенсивність вихідного трафіку, середня довжина вихідного повідомлення та затримка передачі пакету в найбільшій мережі повинні відповідати заданим параметрам: μ =159 кадрів/с.

Для виконання базового налаштування конфігурації пристроїв потрібно враховувати наступні вимоги:

– назви пристроям за правилом: Babenko_type_number;

– пароль *cisco* до консолі і vty;

– пароль *class* до привілейованого режиму;

– усі паролі, що зберігаються у відкритому вигляді, потрібно зашифрувати;

– на усіх лініях vty використання протоколу SSH;

– ім’я користувача та пароль на всіх пристроях за правилом: 123201_Babenko з паролем *admincisco*;

– в якості імені домена потрібно використати ім'я пристрою;

– для шифрування даних потрібно створювати ключ RSA завдовжки 1024 біт;

– для зв’язку між мережами на маршрутизаторах застосувати протокол динамічної маршрутизації OSPF.

При налаштуванні роботи Інтернет в системі необхідно враховувати наступні вимоги:

– потрібно встановити одного провайдера послуг доступу до Інтернет (ISP);

– для виходу робочих станцій в Інтернет необхідно настроїти пограничний маршрутизатор з динамічним NAT за такими даними: ім’я пула: Internet, пул адресів: 209.165.200.5 по 209.165.200.30, номер списку доступу 1;

– потрібно налаштувати сервер HTTP, щоб на вузлах при вводі в рядку браузера http://123.dnipro.ua (http://209.165.200.4) відкривався веб-сайт з відомостями про тему та завдання на кваліфікаційну роботу.

До кожної з підмережі повинно бути під’єднано принаймні 8 комп’ютерів, але у подальшому ця цифра може бути збільшена відповідно до завдання.

2.1.1.2 Вимоги до показників призначення

IoT система МОХ на ЦД має забезпечувати роботу обладнання, а також забезпечувати захист даних, що зберігаються на сервері IoT та додатку від втрат інформації при різного роду помилок чи збоїв.

Крім того, IoT система має забезпечувати захищену передачу даних між підсистемою моніторингу стану здоров’я пацієнта та підсистемою обробки та зберігання даних.

Дані показників призначення повинні відповідати наступному:

– кількість клієнтів, що система може підтримувати одночасно, повинен відповідати значенню у 5000 осіб;

– швидкість реакції системи повинна складати не більше п’яти секунд на відповідь з серверу та не більше 2 хвилин на передачу даних на сервер;

– похибка у точності вимірювання рівню цукру у крові не повинно перевищувати 2%;

– кількість років, скільки потрібно зберігати дані – 45 років.

2.1.1.3 Вимоги до експлуатації, технічого обслуговування, ремонту і збереженню

2.1.1.3.1 Умови і регламент (режим) експлуатації, що повинні забезпечувати використання комплексу технічних засобів (КТЗ) системи із заданими технічними показниками

Вимога до часу експлуатації системи – вона працює цілодобово, 24 години на день та 7 днів на тиждень.

Можливо, також вимкнення сервісів або системи з ціллю заміни або оновлення обладнання, але з попереднім попередженням клієнтів, яке повинно проводитися не менше ніж за годину до вимкнення, або у разі внештатної ситуації – негайно. Попередження надсилати у вигляді електронного листа, повідомлення на телефон, або можливо навіть через служби моніторингу стану здоров’я пацієнта такі як Smart-годинник.

До комплексу технічних засобів системи маються наступні вимоги:

– температура повинна бути у діапазоні від 35 градусів цельсія до 40, але допустиме відхилення на 5 градусів у сторону тепла, а відповідно не менше 35 градусів тепла та не більше 45, для безпечної роботи усіх девайсів та комфорту мобільної групи реагування;

– відносна вологість від 25% до 60%;

– атмосферний тиск від 700мм рт. ст. до 800мм рт. ст.

2.1.1.3.2 Вимоги до параметрів мереж енергопостачання

Живлення IoT серверу та Home Gateway від головної мережі – 230В, 50Гц з заземлювальним контактом, воно повинно обов’язково проходити через мережеві фільтри напруги задля запобігання виходу зі строю дороговартісного

обладнання у разі коливань напруги тощо, також максимальне допустиме відхилення від 230В не повинно перевищувати ± 15В.

Живлення обладнання пацієнта від автономних переносних джерел.

Живлення зв'язку мобільної групи реагування живиться від бортової мережі 12В.

Живлення імплантованого глюкометра використовує енергію від глюкози, які дозволяють пристроям отримувати енергію з глюкози в організмі пацієнта. Це дозволяє пристрою працювати без необхідності заміни батареї.

Живлення Smart годинників відбувається за допомогою літій-іонних батарей номіналом 3,7В.

2.1.1.3.3 Вимоги до кількості, кваліфікації обслуговуючого персоналу і режимам його роботи

Обов'язковою є наявність одного системного адміністратора, що матиме щонайменше бакалаврський ступінь вищої освіти за спеціальністю комп'ютерної інженерії. До його обов'язки належатиме забезпечення підтримки, функціонування та працездатності інформаційних технологій системи, також якщо враховувати що він працює з мережевим обладнанням, він повинен мати досвід роботи у цій сфері та відповідний сертифікат чи документ.

Створення графіку, що передбачав би перевірку та підтримку системним адміністратором працездатності кожного з компонентів системи, а також здійснення позапланових заходів згідно заяв користувачів про некоректну роботу того чи іншого обладнання.

Бажаною є наявність робітника з обслуговування IoT пристроїв, з бакалаврським ступенем вищої освіти за спеціальністю комп'ютерної інженерії. Де до його обов'язків входила б підтримка у робочому стані IoT пристроїв та обслуговування системи.

Режим роботи працівників: виділені вище системний адміністратор та робітник з обслуговування IoT пристроїв лише денна зміна, хоча і можуть бути

терміново викликані у нічні години через критичну для системи помилку або аварію.

2.1.1.3.4 Вимоги до складу, розміщенню та умовам зберігання комплекту запасних деталей, виробів тощо для швидкої заміни у разі позаштатної ситуації

На складі повинні зберігатись речі, потрібні для роботи системи – маршрутизатори Home Gateway, сервер IoT, обладнання для безперебійного живлення серверу у вигляді генератора що покривав би потужність у 500W, датчики-глюкометри та інші IoT пристрої, до складу допускається лише адміністрація та системний адміністратор.

Кількість резервних маршрутизаторів, що зберігаються на складі – 10.

Кількість резервних датчиків глюкометрів, що зберігаються на складі – 100.

Саме складське приміщення повинно бути сухим та закритим для надійного та безпечного зберігання там обладнання.

2.1.1.3.5 Вимоги до регламенту обслуговування

Вимоги до обслуговування обладнання включає в себе графік, де обслуговування проводиться за потреби, але це менше ніж раз на два роки. Сюди входить обслуговування персональних комп’ютерів персоналу, а також комутаторів та маршрутизаторів.

До обслуговування входить: зовнішній і внутрішній огляд пристроїв, перевірка працездатності мережевих фільтрів (заміна у разі несправності за рахунок коштів лікарні), чистка апаратних компонентів, перевірка контактних з’єднань, перевірка параметрів налаштувань працездатності технічних засобів, тестування їх взаємодії, перевірка програмних компонентів на наявність будь-яких вразливостей, неполадок або оновлень, перевірка працездатності мережевих налаштувань та інших необхідних конфігурацій в залежності від типу компонента цілої системи.

2.1.1.3.6 Вимоги до патентної чистоти

Програмні засоби та розроблена структура закладу повинна бути патентною та охоронятись не лише Українським, а й міжнародним законодавством: у разі виявлення використання розроблених технологій без відповідного на то дозволу зі сторони системи – система має повинна мати повне юридичне право подати у відповідний міжнародний суд щодо порушення авторських прав та зтягування компенсації з порушників. Під авторське право будуть внесені написані програми, вироби що використовуються та розроблені саме для цієї галузі.

2.1.1.4 Додаткові вимоги

2.1.1.4.1 Вимоги до системи, пов'язані із особливими умовами її використання

Особливих вимог використання система не має.

2.1.1.4.2 Вимоги до активного обладнання

До активного обладнання відносяться маршрутизатори Home Gateway та сервери, вони повинні забезпечувати стабільну роботу мережі та комунікацію між усіма іншими пристроями, такими як комп'ютери, планшети, різне медичне обладнання, у тому числі між приладами для моніторингу стану пацієнта у вигляді імплантованого глюкометра чи Smart годинника.

Висунуті вимоги до активного обладнання включають в себе наявність щонайменше 1, краще 2 Serial-портів на маршрутизаторі, або 2 Gigabit-Ethernet портів. Це повинно бути високоякісне обладнання від компанії з високою продуктивністю, швидкістю каналу зв'язку 1 Gbit на секунду, а також підтримкою бездротової мережі Wi-Fi для підключення IoT пристроїв.

Сервер повинен бути забезпечений портами Fast Ethernet роз'єму RJ-45 та підтримкою наступних сервісів: HTTP, DNS, TFTP, AAA та IoT для забезпечення виконання задач, що поставлені замовником перед системою.

2.1.1.4.3 Вимоги до комунікаційного обладнання і його розташування

Обладнання такого типу обов’язково повинно бути встановлене у окремому приміщенні з належним рівнем захисту та пожежної безпеки, температура повітря у приміщенні повинна бути у межах від 5 до 30 градусів цельсія, рівень пилозахисту щонайменше IP20, вологість не більше 40%. Доступ до такого приміщення має лише адміністрація та системний адміністратор.

Приміщення повинно бути сухим, закритим та захищеним щонайменше дверним замком. За можливості використовувати камери відеоспостереження з копіюванням відео на сервер лікарні.

2.1.1.4.4 Вимоги до резервування

Обов’язковим до резервування є сервер IoT, на якому зберігається велика кількість даних, що повинні надійно зберігатись.

Три комутатори, що з'єднані між собою з застосуванням агрегування у підмережі MED1, можемо використати для резервування мережі у випадку відмови одного з пристроїв комутування.

2.1.2 Вимоги функцій, виконуваним системою

IoT система медичного обслуговування хворих на цукровий діабет повинна надсилати дані на сервер, зібрані датчиком глюкози, суміжній комп’ютерній системі лікарні. Також вона має отримати рекомендацію від суміжної системи лікарні і переслати цю інформацію до пацієнта, у разі чого отримати від пацієнта запит на виклик МГР та надіслати запит на виклик цій самій групі, отримати від мобільної групи підтвердження що вони готові до виїзду, а по прибутті до точки призначення надіслати звіт системі у форматі виконаних робіт, зміни стану здоров’я пацієнта тощо.

Додаток Health Care аналізує дані, отримані від датчиків, коли система виявляє нестандартну ситуацію, то сповіщення надсилається лікарю у додаток Health Care та інформує про стан здоров’я пацієнта, вказуючи при цьому показники його здоров’я та критичність ситуації.

Очікується, що ця технологія повинна допомогти пацієнтам бути на постійному зв’язку з лікарем. Лікар завжди може віддалено стежити за станом здоров’я своїх пацієнтів, щоб провести профілактичне лікування та втрутитися у екстрених випадках.

Усе обладнання що налаштовується повинно відповідати сучасним потребам та нормам захисту станом на 2024 рік, як самої комп’ютерної системи так і даних що збергіаються та оброблюються нею. Обладнання повинно підтримувати протоколи захисту IPsec, а маршрутизатори підтримувати модуль securityk9 для реалізації технології VPN щоб вбезпечити з’єднання між суміжною та віддаленою мережами.

Обладнання медичного персоналу та групи швидкого реагування повинно бути забезпечено ПЗ Health Care, для моніторингу стану пацієнтів, а також зв’язку з серверами лікарні для ефективності та оперативності роботи.

Лікарня повинна бути забезпечена мережевим обладнанням з налаштованими параметрами захисту та комунікації.

2.1.3 Вимоги до видів забезпечення комп’ютерної системи

2.1.3.1 Вимоги для технічного забезпечення системи

Імплантований глюкометр повинен бути здатен вимірювати рівень глюкози в крові з високою точністю з похибкою не більше 2%, а також мати підтримку оптимізації енергоспоживання за рахунок використання енерегоефективних протоколів.

Датчик фізичної активності у вигляді Smart годинника: Відстежувати фізичну активність пацієнта, включаючи кроки, калорії та серцевий ритм і навантаження на організм.

Планшети або інше комп’ютерне обладнання здатні приймати дані від IoT датчиків через Bluetooth або інші протоколи як Wi-Fi, на якому встановлене ПЗ у вигляді додатку Health Care для моніторингу рівня глюкози, фізичної активності, харчування та інших параметрів повинен бути з функціями нагадувань та оповіщень та підтримкою HL7/FHIR протоколів для обміну даними з

електронними медичними записами (EMR) та іншими медичними інформаційними системами як Helsi та Лікарі 24/7.

Ці вимоги спрямовані на створення ефективної, безпечної та надійної IoT системи для моніторингу та управління станом здоров'я хворих на цукровий діабет.

Обладнання, яке буде використовуватись для впровадження розробленої системи, буде закуповуватись через систему обліку даних – Prozzorro, згідно з технічними вимогами щодо специфікації обладнання заданими до закупівлі. Виробник обладнання та його торгова марка будуть відомі після визначення переможця у тендері.

2.1.3.2 Вимоги до інформаційного забезпечення

Система буде використовувати віддалений IoT сервер, а для його створення буде використовуватись платформа Microsoft Azure, використання рішення забезпечить надійність і доступність, безпеку та масштабованість системи.

Сервер повинен працювати безперервно 24/7, використовуючи служби Azure для забезпечення високої доступності, як-от Azure Availability Zones та Azure Load Balancer. Безпека має відповідати стандартам медичних даних, включаючи HIPAA та GDPR, з використанням шифрування даних в транзиті та на зберіганні, а також багатофакторної аутентифікації.

Сервер повинен приймати дані від IoT пристроїв, таких як глюкометри та Smart годинники, у реальному часі, підтримуючи різні протоколи передачі даних, зокрема MQTT та HTTPS. Для цього використовується Azure IoT Hub, який приймає та управляє повідомленнями від пристроїв. Дані зберігаються у масштабованих базах даних, таких як Azure SQL Database і обробляються в режимі реального часу за допомогою Azure Stream Analytics.

Сповіщення лікарів та пацієнтів про критичні значення показників здійснюється через SMS, електронну пошту або push-сповіщення з використанням Azure Notification Hubs.

Для інтеграції з іншими медичними інформаційними системами система повинна підтримувати стандартні протоколи, такі як FHIR та HL7. Моніторинг та управління системою здійснюються за допомогою Azure Monitor.

2.1.3.3 Вимоги до лінгвистичного забезпечення

Усе ПЗ, що буде так чи інакше використовуватись для роботи персоналу, медичного закладу та системи в цілому, або взаємодії з клієнтом – повинно підтримувати дві мови інтерфейсу: українська та англійська. На сервері повинна бути встановлена ліцензійна Windows Server з останніми оновленнями безпеки.

2.1.3.4 Вимоги до організаційного забезпечення

Системний адміністратор повинен мати фізичний доступ до мережевого обладнання лише після проходження аутентифікації – підтвердження особи. Реалізовано це за допомогою сканера візерунка пальця у кімнаті з мережевим обладнанням.

Кожного разу, як система буде модернізовуватись новими засобами, пристроями, мережевими технологіями тощо, повинні проводитись збори персоналу, де будуть розповідати про зміни та правила поводження з новим обладнанням або використанням тих чи інших технологій, також під час таких зборів будуть робитися бекапи.

2.1.3.5 Вимоги до методичного забезпечення

У системі повинні бути такі документації та інструкції:

– керівництво оператора – ця документація призначена для медичного персоналу, що використовує створену нами систему.

– інструкція з експлуатації – цей документ надає інструкції для фахівців, які використовують систему.

– креслення, схеми і пояснювальна записка, що містять детальну інформацію про технічні аспекти системи. Сюди входить опис архітектури, баз даних, можливих інтеграцій, безпеки та інші технічні деталі.

– FAQ – це книжечка з відповідями на часті запитання, рекомендації щодо усунення проблем та контактні дані служби підтримки медичного закладу.

2.2 Розробка апаратної частини комп’ютерної системи

2.2.1 Розробка загальної архітектури мережі підприємства

Загальна архітектура мережі підприємства включає лише сервіс IoT та HTTP запити до серверу лікарні.

2.2.2 Вибір і обґрунтування структурної схеми комплексу технічних засобів комп’ютерної системи

Комп’ютерна система в галузі допомоги хворим на діабет це комплексна система, що складається з багатьох засобів та рішень, до складу якої входять (рис. 2.1):

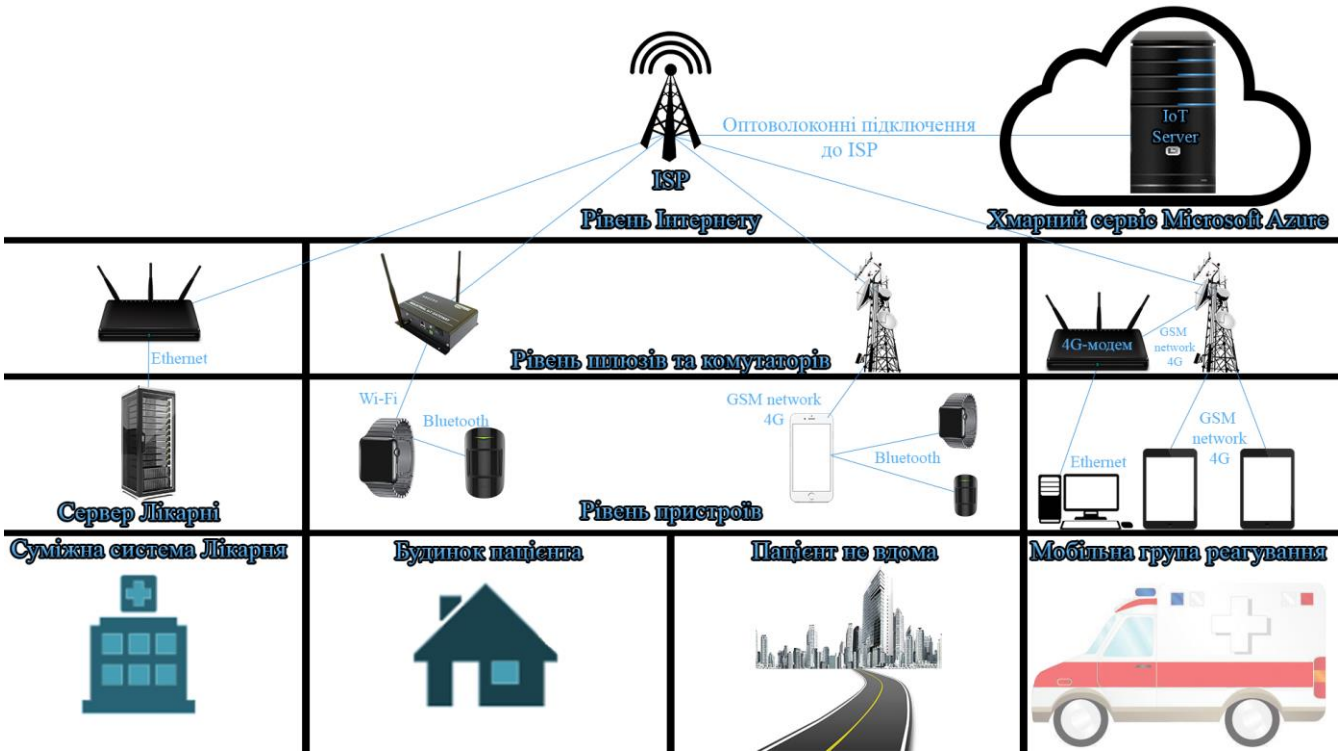

Рисунок 2.1 – Структурна схема комплексу технічних засобів інформаційної системи

2.2.3 Розробка специфікації апаратних засобів комп’ютерної системи

Нижче наведена специфікація обладнання, у кількості мінімально необхідній для працездатності системи.

Починаючи зі Smart-годинника, можна сказати, що він повинен бути з можливістю підключення до зовнішнього глюкометра, що буде імплантовано у пацієнта. У випадку якщо пацієнт відмовляється від імплантації – варіант з інтегрованим у годинник глюкометром. Годинник повинен бути сумісним з iOS та Android, підтримувати Bluetooth та Wi-Fi, щодо автономності пристрою, то вона повинна забезпечувати не менше 5-7 днів роботи від акумулятора в режимі звичайного використання, з підтримкою швидкого заряджання. Важливим параметром годинника є також водонепроникність до 50 метрів (5 ATM) для використання під час плавання та у вологих умовах.

Імплантований глюкометр має компактні розміри, що дозволяє його безпроблемну імплантацію під шкіру, його вага становить приблизно 5 грамів, що мінімізує дискомфорт для користувача. В залежності від моделі, глюкометр може бути обладнаний живленням енергії від глюкози без необхідності заряджати акумулятор ззовні, або вбудованим літій-іонним акумулятором, що забезпечував би безперервну роботу приладу близько 6 місяців без підзарядки, сама підзарядка відбувається бездротовим методом через індукційний зарядний пристрій. Глюкометр оснащений модулем Bluetooth Low Energy (BLE) та NFC для передачі даних на мобільний додаток у режимі реального часу, а вбудована пам'ять дозволяє зберігати до 1000 вимірювань.

Home gateway це маршрутизатор для підключення та управління IoT-пристроїв з моніторингу та догляду за пацієнтами, що має на борту 4 Ethernet порти для підключення до локальної мережі та Інтернету, 1 USB порт для підключення зовнішніх пристроїв та зберігання даних. Підтримуються різноманітні мережеві протоколи, зокрема IPv4/IPv6, DHCP, NAT, VPN, а також HTTPS і SSL/TLS для забезпечення безпеки передаваємих даних. Для комунікації з IoT-пристроями використовуються такі протоколи як MQTT, CoAP та HTTP/HTTPS. Пристрій завжди збирає, зберігає та передає дані від підключених

IoT-пристроїв до IoT-серверу, такі дані як рівень глюкози та дози інсуліну, серцебиття, рівень навантаження. Додатково, наявний LTE модуль, що забезпечує резервне підключення до Інтернету, а також має вбудований акумулятор, який забезпечує до 6 годин автономної роботи у разі блекауту.

Планшети у кількості 2 штук призначені для моніторингу стану здоров'я пацієнтів завдяки встановленому додатку Health Care, ведення історії хвороби пацієнта. Також повинна бути підтримка інтеграції з медичними пристроями через Bluetooth та Wi-Fi для моніторингу рівня глюкози, артеріального тиску та інших показників. Підтримка модулів LTE для мобільного зв'язку та GPS для навігації.

На стаціонарний комп'ютер, яким обладнана мобільна група реагування приходять сповіщення про виклики, а також за його допомогою відбувається звітування щодо виконаних робіт.

Таблиця 2.1 – Специфікація обладнання

Позиція	Найменування і технічна х-ка	Тип, марка, позначення	Одиниця виміру	К-сть	Примітки
1	2	3	4	5	6
1	Smart-годинник	Двоядерний процесор ARM Cortex A53 з частотою 1.15 ГГц, ОЗП 1 ГБ, внутрішня пам'ять 8 ГБ, екран з підтримкою роздільної здатності 360 х 360 пікселів, 1.2 дюйма акумулятор ємністю 450 мАг, підтримка Bluetooth 5.0 Wi-Fi 802.11 b/g/n	шт.	1	Сумісність з iOS та Android. Матеріал: нержавіюча сталь. Розміри: діаметр 42-46 мм, товщина 10-12 мм, вага 40-50 грам.

Продовження таблиці 1.1.

Позиція	Найменування і технічна х-ка	Тип, марка, позначення	Одиниці виміру	К-сть	Примітки
1	2	3	4	5	6
2	Імплантований глюкометр	Розміри: 30 мм х 10 мм х 5 мм, вага 5 грамів, ємність акумулятора: 100 мАг. діапазон вимірювань глюкози: 20 - 600 мг/дл, точність вимірювання: ±5%, Підтримка модулей зв'язку Bluetooth Low Energy (BLE) та NFC	шт.	1	Матеріал корпусу виконаний з біосумісного полімеру, має на борту 4 КБ внутрішньої пам'яті.
3	Home gateway	4xEthernet порти на 10/100/1000 Mbps 1 x USB 3.0 Підтримка Wi-Fi 802.11ac (2.4GHz/5GHz) Bluetooth 5.0 LTE-модуль	шт.	1	Розміри: 200 x 150 x 40 мм Вага: 800 г Споживання: до 25W
4	PC для мобільної групи реагування	Процесор: 6-8 ядерний, 3.4 - 4.5 ГГц; RAM: 16 ГБ; Тип пам'яті: SSD, 240 ГБ або більше; Графічний адаптер: вбудований; Підтримка LAN; Підтримка найновішої ОС: Windows 11.	шт.	1	Для отримання викликів та інформації щодо стану здоров'я пацієнта.
5	Планшет для мобільної групи реагування	Дисплей IPS LCD, 1920x1200 пікселів 8-ми ядерний процесор, 2 ГГц. 8 ГБ ОЗП, вбудована пам'ять 128 ГБ, Android 13. Модулі Wi-Fi, Bluetooth, LTE та GPS.	шт.	2	Наявність швидкої зарядки та батарея ємністю 7000 мАг.

2.2.4 Розрахунок інтенсивності трафіку вихідного трафіку найбільшої локальної мережі підприємства

В підмережі «Зала очікування» встановлений комутатор Cisco2960, що об'єднує до 20 ПК працівників. Вихідний трафік з комутатора Switch1_MED4 надсилається до роутера Router1 в лінію з пропускною здатністю, що становить 1000 Мбіт/с.

Для того, щоб комутатор не був перенасичений, швидкість надходження пакетів не повинна перевищувати швидкості їх відправлення. Вважаємо, що послугами одночасно користуються 100% користувачів. Середня інтенсивність трафіку μ=86 (кадрів/с), а середня довжина повідомлення – 1150 байт.

Теоретично припустимо, що всі користувачі найбільшої підмережі DLS одночасно використовують мережу. В такому разі, пропускна здатність на рівні доступу буде дорівнювати:

$$P_{р.р.} = \mu * L_{пов} * N * 8 = 86 * 1150 * 20 * 8 = 15.8 \text{ Мбіт/с} \quad (2.1)$$

де $L_{пов}$ – середня довжина повідомлення;

N – кількість вузлів в мережі.

Отриманий результат не перевищуватиме заданих параметрів мережі по вихідному каналу, отже перенавантажень не трапиться.

Комутатор SW1_DLS також передає трафік до маршрутизатора зі швидкістю 1000 Мбіт/с. Отже, загальне навантаження на комутатор не повинно перевищувати:

$$\mu_{вих} = 10^9 / (1150 * 8) = 108\,696 \text{ пакетів/с} \quad (2.2)$$

Оскільки в середньому, кожне джерело виробляє 86 пакетів/с, то маршрутизатор обмежений кількістю приєднань, яку ми можемо дізнатись наступним чином:

$$N = \mu_{вих} / \mu = 108\,696/ 86 \approx 1264 \text{ джерела} \quad (2.3)$$

Ця кількість задовольняє кількості вузлів у найбільшій нашій локальній мережі, до якої входить 20 ПК.

Кожен з 20 ПК посилає потік заявок з інтенсивністю у 86 кадрів/с. Звідси, можем розрахувати інтенсивність вихідного трафіку:

$$\lambda = N * \mu = 20*86 = 1720 \text{ пакетів/с.} \quad (2.4)$$

Коефіцієнт затримки:

$$\rho = \frac{\lambda}{\mu вих} = \frac{1720}{108696} = 0{,}016 \quad (2.5)$$

Коефіцієнт зайнятості маршрутизатора:

$$\frac{\rho}{1-\rho} = \frac{0{,}016}{1-0{,}016} = 0{,}016 \quad (2.6)$$

Середня затримка кадру, пов’язана з чергою M/M/1, дорівнює:

$$T = \frac{1}{(\mu-\lambda)} = \frac{1}{(108696 - 1720)} = 9{,}35 \text{ мкс} \quad (2.7)$$

Середня довжина черги:

$$L_{чер} = \frac{\rho^2}{1-\rho} = \frac{0{,}016^2}{1-0.016} = 0.0026 \quad (2.8)$$

Середній час перебування пакета у черзі:

$$T_{оч} = \frac{Lчер}{\lambda} = \frac{0{,}026}{1720} = 0{,}15 \text{ мкс} \quad (2.9)$$

Це значення задовольняє вимогам до затримки в ЛМ.

3 РОЗРОБКА КОРПОРАТИВНОЇ МЕРЕЖІ

3.1 Розрахунок схеми адресації корпоративної мережі

Відповідно до вимог в розділі 2.1.1.1.6 модернізація системи передбачає її інтеграцію у модернізовану КС лікарні, яка має топологію відповідно до завдання на рисунку 1.3.

Мережа складається з 5 підмереж – підрозділів, що повинні містити загалом 321 хост відповідно до завдання замовника (табл. 3.1), а сама мережа повинна мати наступні налаштування: протокол динамічного розподілу IP-адрес DHCP, механізм зміни мережевої адреси NAT, створення віртуального захищеного підключення VPN, протокол динамічної маршрутизації OSPF між мережами, віртуальні локальні мережі VLAN та застосування методу AAA.

Таблиця 3.1 – Кількість хостів в підмережах

MED1	MED2	MED3	MED4	MED5
39	107	24	92	59

Згідно завданню, отриманим від замовника, блок адрес 10.24.8.0/22.

Розробимо цю модель комп’ютерної мережі у програмі Packet Tracer від компанії Cisco, на основі якої нами буде впроваджено IoT систему медичного обслуговування хворих на цукровий діабет.

Використовуючи дані, отримані від замовника, створимо комп’ютерну мережу шляхом розбиття наданого блоку адрес на 5 окремих підмереж. Мережа MED1 є віддаленою та є окремим відділом нашої системи (Технічна підтримка). Також, враховуючи необхідність, потрібно виділити окремі невеликі блоки адрес для зв’язку між самими підмережами, які ми будемо привласнювати підмережам між маршрутизаторами (WAN-підмережі).

Для розбиття мережі на підмережі ми використовували метод маски змінної довжини – VLSM, вона розподіляє IP-адресацію в мережі не використовуючи рамки класової адресації, а це у свою чергу дозволяє нам більш економно розподілити IP-адреси за нашими підмережам та ефективно

використовувати адресний простір IPv4, ніж якби ми стали використовувати маски постійної довжини.

Зробимо розрахунок мережі на підмережі за допомогою вирахування бітів, для цього візьмемо дану нам IP-адресу та переведемо її у бітовий формат, поділивши місця між крапками на октети, та перевівши цифри 10, 24 та 8 у бітовий формат чисел. Так як розрахунок не затроне числа 10 та 24 через невеликі масштаби мережі, то їх можна не чіпати і перевести лише 8 у бітовий формат.

10.24.000010|00.00000000 = 10.24.8.0 /22

Для розрахунку методом VLSM необхідно відсортувати мережі за розміром за спаданням. Почнемо розрахунок з найбільшої мережі MED2, визначаємо кількість бітів що потрібно буде відділити від основної частини, якщо потрібно 107 хостів, то найближче доступне та більше значення це 127, якщо рахувати побітово то це 7 бітів. Проводимо розрахунок від першої адреси до останньої.

MED2 – 107 – 7 біт:

10.24.00001000.0|0000000 /25 – 10.24.8.0/25 – номер мережі;

10.24.00001000.0|0000001 /25 – 10.24.8.1/25 – перша допустима адреса;

10.24.00001000.0|1111111 /25 – 10.24.8.126/25 – остання допустима адреса.

MED4 – 92 – 7 біт:

10.24.00001000.1|0000000/25 – 10.24.8.128/25 – номер мережі;

10.24.00001000.1|0000001/25 – 10.24.8.129/25 – перша допустима адреса;

10.24.00001000.1|1111110/25 – 10.24.8.254/25 – остання допустима адреса.

MED5 – 59 – 6 біт:

10.24.00001001.01|000000/26 – 10.24.9.0/26 – номер мережі;

10.24.00001001.01|000000/26 – 10.24.9.1/26 – перша допустима адреса;

10.24.00001001.01|111111/26 – 10.24.9.62/26 – остання допустима адреса.

MED1 – 39 – 6 біт:

10.24.00001001.01|000000/26 – 10.24.9.64/26 – номер мережі;

10.24.00001001.01|000000/26 – 10.24.9.65/26 – перша допустима адреса;

10.24.00001001.01|111111/26 – 10.24.9.126/26 – остання допустима адреса.

MED3 – 24 – 5 біт:

10.24.00001001.100|00000/27 – 10.24.9.128/27 – номер мережі;

10.24.00001001.100|00000/27 – 10.24.9.129/27 – перша допустима адреса;

10.24.00001001.100|11111/27 – 10.24.9.158/27 – остання допустима адреса.

Розрахувавши мережу таким чином, у таблиці 3.2 можна побачити результат розподілу адрес для нашої майбутньої корпоративної мережі, яку ми впровадимо у нашій роботі.

Таблиця 3.2 – Розрахунок адресації корпоративної мережі

Назва мережі	К-сть вузлів, доступно/ потрібно	Номер мережі	Маска мережі	Початкове значення діапазону можливих адрес вузлів у підмережі	Кінцеве значення діапазону можливих адрес вузлів у підмережі
MED2	126/107	10.24.8.0	/25	10.24.8.1	10.24.8.126
MED4	126/92	10.24.8.128	/25	10.24.8.129	10.24.8.254
MED5	62/59	10.24.9.0	/26	10.24.9.1	10.24.9.62
MED1	62/39	10.24.9.64	/26	10.24.9.65	10.24.9.126
MED3	30/24	10.24.9.128	/27	10.24.9.129	10.24.9.158
WAN 1	2/2	10.0.1.0	/30	10.0.1.1	10.0.1.2
WAN 2	2/2	10.0.1.4	/30	10.0.1.5	10.0.1.6
WAN 3	2/2	10.0.1.8	/30	10.0.1.9	10.0.1.10
WAN 4	2/2	10.0.1.12	/30	10.0.1.13	10.0.1.14
WAN 5	2/2	10.0.1.16	/30	10.0.1.17	10.0.1.18
ISP 1	2/2	209.165.202.0	/30	209.165.202.1	209.165.202.2
ISP 2	2/2	64.100.13.0	/30	64.100.13.1	64.100.13.2

Також до таблиці було включено адресацію підмереж провайдера, що являє собою зв’язок між віддаленою мережею та мережею лікарні, і використовується у подальшому при моделюванні мережі. Крім того, було розроблено схему IP-адресації між маршрутизаторами у нашій корпоративній мережі лікарні, користуючись тим самим методом, що і при розрахунку адрес для підмереж.

Згідно з завданням, для зв’язку між маршрутизаторами нами було використано блок адрес 10.0.1.0/24. Розраховані адреси підмережі, адреси зв’язку маршрутизаторів у мережі, а також адреси зв’язку мережі лікарні з провайдером та провайдера з віддаленою мережею можна побачити у таблиці 3.2.

3.2 Розрахунок схеми адресації пристроїв

Після розрахунку IP-адресації підмереж лікарні, була виконана робота з привласнення мережевої адреси для інтерфейсів маршрутизаторів мережі лікарні, віддаленій мережі MED1 та маршрутизатору провайдера, враховуючи налаштування, що будуть нами виконані у наступних підрозділах, розподіл адрес інтерфейсам маршрутизаторів приведено у таблиці 3.4. За вимогами замовника, перші адреси з підмереж були привласнені саме тим інтерфейсам маршрутизаторів, що ведуть до підмереж, тим самим утворюючи Default Gateway.

Таблиця 3.1 – Адресації інтерфейсів пристроїв

Пристрій	Інтерфейс	IP-адреса	Маска
ISP	Se0/3/1	209.165.202.2	255.255.255.252
	Gig0/0	64.100.13.1	255.255.255.252
	Gig0/1	209.165.201.1	255.255.255.240
Router1	Se0/3/0	10.0.1.1	255.255.255.252
	Se0/3/1	10.0.1.5	255.255.255.252
	Gig0/0	10.0.1.9	255.255.255.252
	Gig0/1	10.24.8.129	255.255.255.128
Router2	Se0/2/0	10.0.1.18	255.255.255.252
	Se0/3/0	10.0.1.2	255.255.255.252
	Se0/3/1	10.0.1.6	255.255.255.252
	Gig0/0	10.24.9.1	255.255.255.192
	Gig0/1	10.24.8.1	255.255.255.128
Router3	Se0/2/0	10.0.1.17	255.255.255.252
	Se0/3/0	10.0.1.14	255.255.255.252
	Se0/3/1	209.165.202.1	255.255.255.252
Router4	Se0/3/0	10.0.1.13	255.255.255.252
	Gig0/0	10.0.1.10	255.255.255.252
	Gig0/1	10.24.9.129	255.255.255.224
Router5	Gig0/0	64.100.13.2	255.255.255.252
	Gig0/1	10.24.9.65	255.255.255.192

Наступним кроком нами було розраховано та вручну, статично розподілено IP–адреси серверів лікарні (табл. 3.5) за умовою, що вони отримуватимуть адресу

за формулою перша адреса підмережі + 9 + №_варіанту з допустимих IP-адрес підмережі, у якій вони знаходяться, відповідно якщо перша допустима адреса це 10.24.9.129 тоді додаємо 9 і по номеру варіанта 1, виходить 10.24.9.139 це адреса сервера.

Таблиця 3.2 – Адресація інтерфейсів серверів

Сервер	Інтерфейс	IP–адреса	Префікс	Шлюз
Server HTTP	Fa0	10.24.9.139	27	10.24.9.129
Server DNS	Fa0	10.24.9.140	27	10.24.9.129
Server TFTP	Fa0	10.24.9.11	26	10.24.9.1

Наостанок, було виконано роботу з розподілу IP-адреси SVI-інтерфейсам на комутаторах у підмережах (табл. 3.6), котрі, згідно за вимогою замовника, отримали IP-адреси, починаючи с другої допустимої адреси у пулі своєї підмережі.

Таблиця 3.3 – Схема адресації пристроїв для комутації

Пристрій	Інтерфейс	IP-адреса	Маска	Шлюз
Sw01_ MED2	vlan1	10.24.8.2	255.255.255.128	10.24.8.1
Sw00_ MED2	vlan1	10.24.8.3	255.255.255.128	10.24.8.1
Sw02_ MED2	vlan1	10.24.8.4	255.255.255.128	10.24.8.1
Sw1_MED5	vlan1	10.24.9.2	255.255.255.192	10.24.9.1
Sw1_MED4	vlan1	10.24.8.130	255.255.255.128	10.24.8.130
Sw1_MED3	vlan1	10.24.9.128	255.255.255.224	10.24.9.129
Sw03_MED1	vlan1	10.24.9.66	255.255.255.192	10.24.9.65
Sw04_ MED1	vlan1	10.24.9.67	255.255.255.192	10.24.9.65
Sw05_ MED1	vlan1	10.24.9.68	255.255.255.192	10.24.9.65

3.3 Розробка моделі корпоративної мережі

Розроблену схему корпоративної мережі «Центру піклування про здоров’я пацієнтів хворих на цукровий діабет» та розраховану схему адресації пристроїв і підмереж, що представлені у таблицях 3.2 – 3.6, було здійснено роботу по втіленню їх у логічну топологію для перевірки на моделі комп’ютерної системи

у програму Cisco Packet Tracer. Результат створення логічної топології мережі можна побачити на рисунку 3.1.

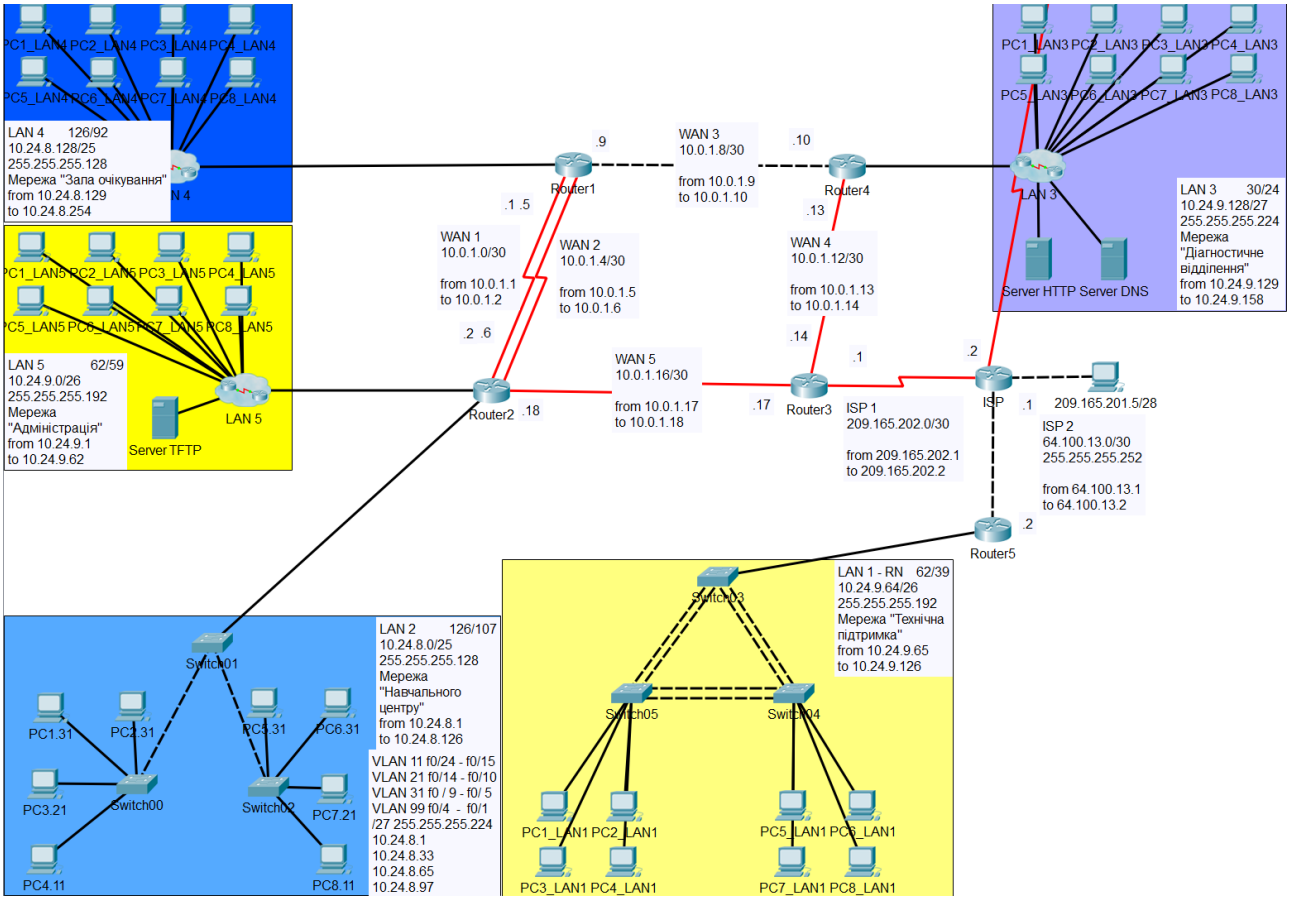

Рисунок 3.1 – Логічна топологія корпоративної мережі

3.4 Базове налаштування та конфігурація пристроїв

Перед впровадженням основних налаштувань з адресації та різних протоколів, було виконано роботу з базових налаштувань пристроїв, таких як комутатори та маршрутизатори, було налаштовано назву пристрою відповідно до завдання, встановлення паролю до консольної лінії 0, до ліній VTY 0 15, встановлення паролю до привілейованого режиму, увімкнення базового шифрування паролів, налаштування вітального повідомлення при увімкненні консолі на пристрої, встановлення доменного імені, створення локального користувача, а також шифрування паролів за допомгою протколу SSH.

Приклад налаштування пристроїв наведено на рисунку у вигляді налаштування маршрутизатора Babenko_Router_3:

```
Router>! vkhid do pryveleiovanoho rezhymu korystuvacha
Router>enable
Router#! vkhid u rezhym hlobalnoi konfihuratsii
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#! pryvlasnennia imeni prystroiu
Router(config)#hostname Babenko_Router_3
Babenko_Router_3(config)#! vstanovlennia paroliu do konsolnoi linii
Babenko_Router_3(config)#line console 0
Babenko_Router_3(config-line)#password cisco
Babenko_Router_3(config-line)#login
Babenko_Router_3(config-line)#! vstanovlennia paroliu do linii vty
Babenko_Router_3(config-line)#line VTY 0 15
Babenko_Router_3(config-line)#password cisco
Babenko_Router_3(config-line)#login
Babenko_Router_3(config-line)#! vstanovlennia paroliu dlia vkhodu u pryveleiovanyi rezhym korystuvacha
Babenko_Router_3(config-line)#enable secret class
Babenko_Router_3(config)#! vmykannia tekhnolohii bazovoho shyfruvannia paroliv
Babenko_Router_3(config)#service password-encryption
Babenko_Router_3(config)#! Nalashtuvannia pryvitalnoho povidomlennia pry zapusku konsoli na prystroi
Babenko_Router_3(config)#banner motd Attention! BabenkoOlA
Babenko_Router_3(config)#
Babenko_Router_3(config)# ! Vstanovlennia domennoho imeni
Babenko_Router_3(config)#ip domain-name Babenko_Router_3
Babenko_Router_3(config)#! vmykannia shyfruvannia paroliv za protokolom rsa
Babenko_Router_3(config)#crypto key generate rsa
The name for the keys will be: Babenko_Router_3.Babenko_Router_3
Choose the size of the key modulus in the range of 360 to 2048 for your
  General Purpose Keys. Choosing a key modulus greater than 512 may take
  a few minutes.

How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

Babenko_Router_3(config)#! Stvorennia lokalnoho korystuvacha
*мар 1 0:0:20.129: %SSH-5-ENABLED: SSH 1.99 has been enabled
Babenko_Router_3(config)#username 123201_Babenko password admincisco
Babenko_Router_3(config)#line vty 0 15
Babenko_Router_3(config-line)#! uvimkneno vykorystannia protokolu SSH na vsikh VTY liniiakh
Babenko_Router_3(config-line)#transport input ssh
```

Рисунок 3.2 – Базові налаштування на Babenko_Router_3

3.5 Налаштування маршрутизаторів корпоративної мережі

Переходимо до основної частини налаштування мережі – налаштування адресації пристроїв у мережі.

Сценарій наступний:

– налаштування інтерфейсів;

– налаштування протоколу маршрутизації;

– оголошення на граничному маршрутизаторі маршруту за замовчуванням і розповсюдження його через повідомлення OSPF;

– налаштування маршрутизатора в MED1 в ролі сервера DHCP для хостів в локальній мережі;

– налаштування сервера та служб AAA.

Перш за все, було зроблено налаштування інтерфейсів маршрутизаторів на кожному із задіяних інтерфейсів. На кожному з маршрутизаторів у мережі було привласнено свою власну IP-адресу та маску по таблиці 3.3, а на всіх Serial-інтерфейсах було встановлено тактову частоту, затримку та пропускну спроможність як на рисунку :

```
Babenko_Router_3(config)#   interface Serial0/3/1
Babenko_Router_3(config-if)#ip address 209.165.202.1 255.255.255.252
Babenko_Router_3(config-if)#clock rate 128000
Babenko_Router_3(config-if)#delay 7500
Babenko_Router_3(config-if)#bandwidth 128
Babenko_Router_3(config-if)#
```

Рисунок 3.3 – Налаштування Serial-інтерфейсу

Задля реалізації маршрутизації було обрано протокол динамічної маршрутизації OSPF. Який заснований на технології відстеження стану каналу, що використовує Алгоритм Дейкстри для знаходження найкоротшого шляху. Він поширює інформацію про доступні маршрути між маршрутизаторами однієї автономної системи.

Говорячи про цей протокол, можна виділити основні переваги, а саме: відсутність обмежень досяжності, оптимальне використання пропускної здатності мережі, підтримка масок змінної довжини VLSM та оптимальний вибір шляху маршрутизації, якраз те що нам і потрібно.

Приклад налаштування на Babenko_Router_3 представлено на рисунку 3.4. Він буде анонсувати три мережі в протоколі OSPF: 10.0.1.12/30, 10.0.1.16/30 та 209.165.202.0/30. Це дозволить іншим маршрутизаторам в OSPF-домені дізнатися про ці мережі та відповідні шляхи до них.

```
Babenko_Router_3(config)#router ospf 1
Babenko_Router_3(config-router)#network 10.0.1.12 0.0.0.3 area 0
Babenko_Router_3(config-router)#network 10.0.1.16 0.0.0.3 area 0
Babenko_Router_3(config-router)#network 209.165.202.0 0.0.0.3 area 0
Babenko_Router_3(config-router)#
```

Рисунок 3.4 – Налаштування OSPF на Babenko_Router_3

Після активації протоколу на двох маршрутизаторах та оголошення спільної мережі, маршрутизатори стають «сусідами», та починають

обмінюватись таблицею маршрутизації (рис. 3.5). Дана інформація показує, що у маршрутизатора Babenko_Router_3 встановлено сусідство за протоколом OSPF з трьома іншими пристроями в мережі. Це свідчить про те, що налаштування OSPF на Babenko_Router_3 виконано коректно.

```
Babenko_Router_3#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
10.24.9.1         0   FULL/  -        00:00:30    10.0.1.18       Serial0/2/0
10.24.9.129       0   FULL/  -        00:00:30    10.0.1.13       Serial0/3/0
209.165.202.2     0   FULL/  -        00:00:39    209.165.202.2   Serial0/3/1
```

Рисунок 3.5 – Таблиця сусідніх пристроїв OSPF на Babenko_Router_3

Так як за замовчуванням протокол використовує поширення повідомлень визначений період часу, потрібно вимкнути це автоматичне поширення та зробити його лише на тих інтерфейсах, що йдуть до маршрутизаторів (рис. 3.6). Також зробимо налаштування пропускної спроможності між маршрутизаторами. OSPF вартість інтерфейсу розраховується на основі його пропускної здатності. За замовчуванням OSPF використовує еталонну пропускну здатність 100 Мбіт/с. Тут ми змінюємо її на 1000 Мбіт/с, що впливає на розрахунок витрат OSPF (рис. 3.6).

```
Babenko_Router_3(config)#router ospf 1
Babenko_Router_3(config-router)#passive-interface default
Babenko_Router_3(config-router)#no passive-interface Serial0/2/0
Babenko_Router_3(config-router)#no passive-interface Serial0/3/0
Babenko_Router_3(config-router)#no passive-interface Serial0/3/1
Babenko_Router_3(config-router)#auto-cost reference-bandwidth 1000
```

Рисунок 3.6 – Керування поведінкою OSPF на певних інтерфейсах

Тепер налаштуємо статичні маршрути на Babenko_Router_3, так як через цей маршрутизатор забезпечується під’єднання до Інтернет-провайдера. Було налаштовано лише 3 статичні маршрути, один за замовчуванням, що веде до провайдера у випадку, якщо маршрутизатор отримує пакет, мережу призначення якого він не знає, другий що веде до підмережі комп’ютера який під’єднано до ISP і останній це NAT адрес, щоб коли пакет йшов назад роутер знав цю мережу та куди відправляти.

Babenko_Router_3 (config)# ip route 0.0.0.0 0.0.0.0 209.165.202.2

Babenko_Router_3 (config)# ip route 209.165.201.0 255.255.255.240 209.165.202.2

Babenko_Router_3 (config)# ip route 209.165.200.32 255.255.255.224 209.165.202.2

Тепер виконаємо налаштування служби AAA (authentication, authorization, accounting), який використовується для опису процесу надання доступу до комп'ютерної мережі та контролю за ним.

Для налаштування цього сервісу потрібно буде зробити налаштування як на маршрутизаторі, так і на будь-якому сервері, де буде увімкнена служба AAA.

Почнімо з маршрутизатора (рис. 3.7).

```
Babenko_Router_3(config)#! uvimknennia sluzhby
Babenko_Router_3(config)#aaa new-model
Babenko_Router_3(config)#aaa authentication login console group radius local
Babenko_Router_3(config)#line console 0
Babenko_Router_3(config-line)#! metod autentyfikatsii dlja konsoli
Babenko_Router_3(config-line)#login authentication console
Babenko_Router_3(config-line)#aaa authentication login default local
Babenko_Router_3(config)#! stvorennia lokalnoi bazy danykh korystuvacha
Babenko_Router_3(config)#username 123201_Babenko password admin123
Babenko_Router_3(config)#line vty 0 15
Babenko_Router_3(config-line)#login authentication default
Babenko_Router_3(config-line)#exit
```

Рисунок 3.7 – Налаштування AAA на маршрутизаторі

Наступним кроком є налаштування самого серверу, де впроваджено цю службу, було обрано HTTP сервер для впровадження служби AAA (рис. 3.8).

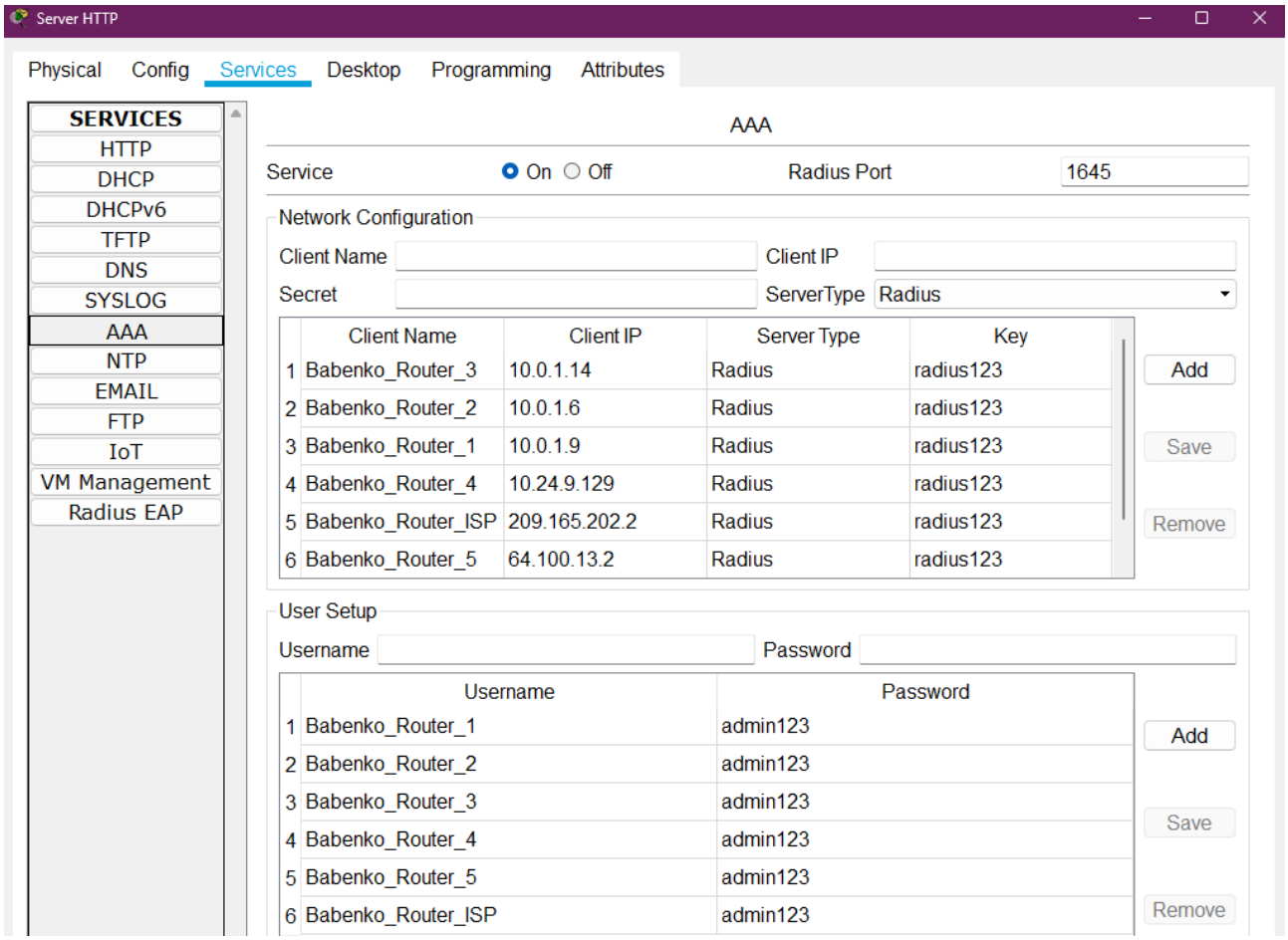

Рисунок 3.8 – Налаштування RADIUS-сервера

Як бачимо з рисунку 3.8, було створено базу даних користувачів, що мають власне ім'я та пароль для логіну. Для заповнення рядків Client IP було використано адресу інтерфейсу пристрою, яка є у напрямку RADIUS-серверу.

На останок, на маршрутизаторах у мережі було впроваджено динамічне розподілення адрес пристроям, що належать до підмережі, завдяки протоколу DHCP. Пристрій буде отримувати одну адресу з виділеного пулу адрес. На рисунку 3.9 приклад налаштування DHCP на Babenko_Router_5. Метою цієї конфігурації є створення пулу DHCP під назвою "poollan1", який призначає IP-адреси з мережі 10.24.9.64/26, встановлю IP-адресу шлюзу за замовчуванням на 10.24.9.65, IP-адресу DNS-сервера на 10.24.9.140 та виключає діапазон IP-адрес із призначення клієнтам.

```
Babenko_Router_5(config)#ip dhcp pool poollan1
Babenko_Router_5(dhcp-config)#! merezha pulu
Babenko_Router_5(dhcp-config)#network 10.24.9.64 255.255.255.192
Babenko_Router_5(dhcp-config)#! marshrut za zamovchuvanniam, yakyi budut otrymuvaty prystroi
Babenko_Router_5(dhcp-config)#default-router 10.24.9.65
Babenko_Router_5(dhcp-config)#! adresa DNS-serveru, yaku budut otrymuvaty prystroi
Babenko_Router_5(dhcp-config)#dns-server 10.24.9.140
Babenko_Router_5(dhcp-config)#exit
Babenko_Router_5(config)#! vykliuchennia z pulu 10ty adres, zrobleno tse zadlia rezervuvannia marshrutyzatsii
Babenko_Router_5(config)#ip dhcp excluded-address 10.24.9.65 10.24.9.74
Babenko_Router_5(config)#
```

Рисунок 3.9 – Налаштування на Babenko_Router_5 служби DHCP

Для перевірки, що служба DHCP на маршрутизаторі працює і виділяю з пулу IP-адреси, застосуємо команду «show ip dhcp binding», яка відображає поточні прив'язки DHCP на маршрутизаторі (рис. 3.10). Результат показує, що наразі до маршрутизатора прив'язано 8 клієнтів DHCP з IP-адресами в діапазоні від 10.24.9.75 до 10.24.9.82. Кожен клієнт має унікальну апаратну адресу, пов'язану з його IP-адресою.

```
Babenko_Router_5#sh ip dhcp binding
IP address       Client-ID/              Lease expiration        Type
                 Hardware address
10.24.9.75       0090.216B.E58D           --                     Automatic
10.24.9.76       0006.2A93.29B9           --                     Automatic
10.24.9.78       00D0.BCAA.2D89           --                     Automatic
10.24.9.77       0001.9758.B5B4           --                     Automatic
10.24.9.79       0007.EC29.45A6           --                     Automatic
10.24.9.80       000D.BD0A.1370           --                     Automatic
10.24.9.81       0007.EC85.DDE8           --                     Automatic
10.24.9.82       0060.2F0D.004E           --                     Automatic
```

Рисунок 3.10 – Таблиця прив'язок DHCP

3.6 Налаштування доступу в Інтернет

Задля імітації справжньої мережі, було додано провайдера послуг Інтернет, у вигляді маршрутизатора ISP. Для маршрутизації пристроїв з підмереж нашої корпоративної мережі до мережі Інтернет використовується динамічний NAT, який налаштовано на граничних маршрутизаторах з ISP.

Для основної групи підмереж, що пов'язані з пристроєм Babenko_Router3 і який у свою чергу є граничним до ISP було використано пул адрес з 209.165.200.5 по 209.165.200.30 згідно завдання.

Для реалізації NAT ми використаємо розширений список доступу, що забороняв би трафік з основної мережі до віддаленої тMED1 та дозволяв увесь інший (рис. 3.11).

```
Babenko_Router_3(config)#ip access-list extended NAT
Babenko_Router_3(config-ext-nacl)#! zaborona trafiku z pidmerezhi MED2 do pidmerezhi MED1
Babenko_Router_3(config-ext-nacl)#deny ip 10.24.8.0 0.0.0.127 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#! zaborona trafiku z pidmerezhi MED3 do pidmerezhi MED1
Babenko_Router_3(config-ext-nacl)#deny  ip 10.24.9.128 0.0.0.31 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#! zaborona trafiku z pidmerezhi MED4 do pidmerezhi MED1
Babenko_Router_3(config-ext-nacl)#deny ip 10.24.8.128 0.0.0.127 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#! zaborona trafiku z pidmerezhi MED5 do pidmerezhi MED1
Babenko_Router_3(config-ext-nacl)#deny ip 10.24.9.0 0.0.0.63 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.8.0 0.0.0.127 any
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.9.128 0.0.0.31 any
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.8.128 0.0.0.127 any
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.9.0 0.0.0.63 any
Babenko_Router_3(config-ext-nacl)#
```

Рисунок 3.11 – Створення ACL для NAT

Тепер, використовуючи новостворений список доступа ми створимо пул Internet та поєднаємо його зі списком NAT. Ключове слово «inside» вказує, що функція NAT має перекладати вихідні IP-адреси трафіку, що надходить із внутрішньої мережі, а ключове слово «source» вказує, що переклад має відбуватися для вихідного трафіку.

```
Babenko_Router_3(config)#
Babenko_Router_3(config)#ip nat pool Internet 209.165.200.5 209.165.200.30
netmask 255.255.255.224
Babenko_Router_3(config)#ip nat inside source list NAT pool Internet
Babenko_Router_3(config)#
```

Рисунок 3.12 – Налаштування NAT

Останнім кроком у налаштуванні NAT є вказання ролей інтерфейсів у роботі NAT, для цього ми заходимо у конфігурацію інтерфейсу та вказуємо чи є він вхідним або ж вихідним (рис. 3.13).

```
Babenko_Router_3(config)#interface Serial0/3/1
Babenko_Router_3(config-if)# ip nat outside
Babenko_Router_3(config-if)# interface Serial0/3/0
Babenko_Router_3(config-if)# ip nat inside
Babenko_Router_3(config-if)# interface Serial0/2/0
Babenko_Router_3(config-if)# ip nat inside
```

Рисунок 3.13 – Налаштування інтерфейсів NAT

Таким чином, вважаємо налаштування NAT завершеними. Але у нас залишається ще один граничний з ISP маршрутизатор під назвою Router5, на

ньому ми робимо аналогічні дії, але з урахуванням іншого пулу адресів та іншої адресації у списку доступу. Пул адрес з 209.165.200.37 по 209.165.200.62.

Згадуючи раніше заборонену комунікацію між віддаленою підмережею та іншими підмережами, ми виконаємо налаштування безпечного з'єднання між ними за допомгою VPN. Для реалізації цього з'єднання нам потрібно створити розширений список доступу, де потрібно дозволити трафік з підмереж до віддаленої підмережі.

```
Babenko_Router_3(config)#ip access-list extended VPN1
Babenko_Router_3(config-ext-nacl)#! dozvil na prokhodzhennia trafiku z MED2 do MED1
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.8.0 0.0.0.127 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#! dozvil na prokhodzhennia trafiku z MED3 do MED1
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.9.128 0.0.0.31 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#! dozvil na prokhodzhennia trafiku z MED4 do MED1
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.8.128 0.0.0.127 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#! dozvil na prokhodzhennia trafiku z MED5 do MED1
Babenko_Router_3(config-ext-nacl)#permit ip 10.24.9.0 0.0.0.63 10.24.9.64 0.0.0.63
Babenko_Router_3(config-ext-nacl)#
```

Рисунок 3.14 – Налаштування ACL для VPN

Далі, потрібно активувати на кожному маршрутизаторі особливий модуль securityk9 та погодитись з умовами його використання.

```
Babenko_Router_3(config)#license boot module c2900 technology-
package securityk9
```

Погоджуємось, зберігаємо конфігураційний файл та перезавантажуємо маршрутизатори, після цих дій модуль стає активним і ми можемо приступати до створення VPN-з'єднання, додаємо властивості криптографічної політики ISAKMP 10 та створюємо загальний ключ шифрування babenko.

```
Babenko_Router_3(config)#crypto isakmp policy 10
Babenko_Router_3(config-isakmp)#encryption aes
Babenko_Router_3(config-isakmp)#authentication pre-share
Babenko_Router_3(config-isakmp)#group 2
Babenko_Router_3(config-isakmp)#crypto isakmp key babenko address 64.100.13.2
Babenko_Router_3(config)#crypto map MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
        and a valid access list have been configured.
Babenko_Router_3(config-crypto-map)#description VPN connection to R5
Babenko_Router_3(config-crypto-map)#set peer 64.100.13.2
Babenko_Router_3(config-crypto-map)#set transform-set OLA
ERROR: transform set with tag OLA does not exist.
Babenko_Router_3(config-crypto-map)#match address VPN1
Babenko_Router_3(config-crypto-map)#interface Serial0/3/1
Babenko_Router_3(config-if)#crypto map MAP
*Jan  3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
```

Рисунок 3.15 – Налаштування VPN на Babenko_Router_3

Останніми двома командами було прив’язано створене криптографічне зіставлення MAP до вихідного інтерфейсу маршрутизатора Router3, після чого ми бачимо наступне повідомлення, що повідомляє про успішне налаштування:

*Jan 3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON

Аналогічні симетричні налаштування необхідно виконати на протилежному кінці тунеля VPN на маршрутизаторі babenko_Router_5.

3.7 Захист інформації від несанкціонованого доступу

Для захисту інформації в системі від несанкціонованого доступу та розділення користувачів підмережі на певну кількість підрозділів за виконуваними ними функціями, мережу MED2 було розділено на 4 частини (VLAN), три з яких є основними та призначені для користувачів, а четверта створена для призначення інтерфейсам пристроїв комутації та має назву Management (рис. 3.16).

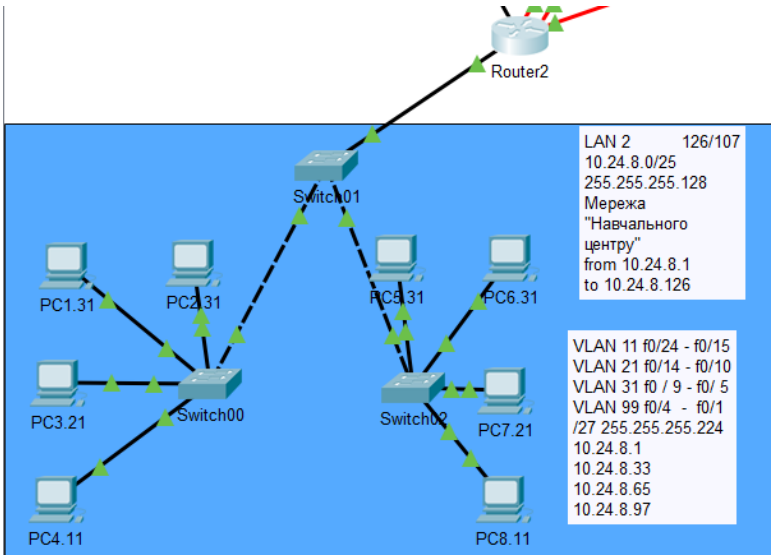

Рисунок 3.16 – Вигляд підмережі MED 2

Для кожного сегменту буде застосовуватися свій адресний блок, тому мережу MED2 необхідно буде поділити на 4 рівні підмережі. А таблиці 3.4 результат поділу.

Таблиця 3.4 – Адресація мереж VLAN

Назва мережі	VLAN ID	К-сть вузлів	Номер мережі	Маска мережі	Перший можливий	Останній можливий
Accounting	11	32/30	10.24.8.0	/27	10.24.8.1	10.24.8.30
Resources	21	32/30	10.24.8.32	/27	10.24.8.33	10.24.8.62
Guest	31	32/30	10.24.8.64	/27	10.24.8.65	10.24.8.94
Management	99	32/30	10.24.8.96	/27	10.24.8.97	10.24.8.126

Також, варто зазначити що VLAN`и розподілені за портами FastEthernet по таблиці 3.5.

Таблиця 3.5 – Список мереж VLAN «Навчального центру»

Номер VLAN	Ім’я VLAN	Примітка	Розподілення портів
1	Default	Не використовується	-
11	Accounting	Відділ кадрів	f0/15-24
21	Resources Department	Відділ зарплатних проектів	f0/10-14
31	Guest	Відділ касових операцій	f0/5-9
99	Management	Для управління пристроями	-
100	Native	Власна	f0/1-4

Створення VLAN та привласнення їм імен на прикладі центрального комутатору підмережі MED2 на рисунку 3.17.

```
Babenko_Switch_01(config)#vlan 11
Babenko_Switch_01(config-vlan)#name Accounting
Babenko_Switch_01(config-vlan)#vlan 21
Babenko_Switch_01(config-vlan)#name Resources_Department
Babenko_Switch_01(config-vlan)#vlan 31
Babenko_Switch_01(config-vlan)#name Guest
Babenko_Switch_01(config-vlan)#vlan 99
Babenko_Switch_01(config-vlan)#name Management
Babenko_Switch_01(config-vlan)#vlan 100
Babenko_Switch_01(config-vlan)#name Native
Babenko_Switch_01(config-vlan)#
```

Рисунок 3.17 – Оголошення VLAN

Далі, розглядаючи налаштування на прикладі центрального комутатора підмережі, було привласнено IP-адресу на SVI-інтерфейс vlan 99 з мережі 10.24.8.96 що показана у таблиці 3.4 (рис. 3.18).

```
Babenko_Switch_01(config)#interface vlan 99
Babenko_Switch_01(config-if)#ip address 10.24.8.98 255.255.255.224
Babenko_Switch_01(config-if)#ip default-gateway 10.24.8.97
```

Рисунок 3.18 – Привласненя IP-адреси на SVI-інтерфейсі

Наступним кроком заходимо на інтерфейс комутатора що веде до маршрутизатора та робимо налаштування дозволів порта (рис. 3.19).

```
Babenko_Switch_01(config)#interface g0/1
Babenko_Switch_01(config-if)#switchport mode trunk
Babenko_Switch_01(config-if)#switchport trunk native vlan 100
Babenko_Switch_01(config-if)#switchport trunk allowed vlan 11, 21, 31, 99, 100
```

Рисунок 3.19 – Налаштування транкових потрів

Далі ми будемо робити розподіл інтерфейсів за VLAN`ами та змінимо режим роботи інтерфейсів (рис. 3.20).

```
Babenko_Switch_01(config)#interface range f0/5-9
Babenko_Switch_01(config-if-range)#switchport mode access
Babenko_Switch_01(config-if-range)#switchport access vlan 31
Babenko_Switch_01(config-if-range)#interface range f0/10-14
Babenko_Switch_01(config-if-range)#switchport mode access
Babenko_Switch_01(config-if-range)#switchport access vlan 21
Babenko_Switch_01(config-if-range)#interface range f0/15-24
Babenko_Switch_01(config-if-range)#switchport mode access
Babenko_Switch_01(config-if-range)#switchport access vlan 11
Babenko_Switch_01(config-if-range)#int range f0/1-4
Babenko_Switch_01(config-if-range)#switchport mode trunk
Babenko_Switch_01(config-if-range)#switchport trunk native vlan 100
```

Рисунок 3.20 – Налаштування портів доступу

Переходимо до наступного етапу налаштування, де налаштовуватимемо маршрутизатор до якого під’єднано мережу VLAN, конкретно до маршрутизатора Router2, почнімо з налаштування DHCP (рис. 3.21):

```
Babenko_Router_2(config)#ip dhcp excluded-address 10.24.9.1 10.24.9.10
Babenko_Router_2(config)#ip dhcp excluded-address 10.24.9.11
Babenko_Router_2(config)#ip dhcp excluded-address 10.24.8.1 10.24.8.10
Babenko_Router_2(config)#ip dhcp excluded-address 10.24.8.33 10.24.8.42
Babenko_Router_2(config)#ip dhcp excluded-address 10.24.8.65 10.24.8.74
Babenko_Router_2(config)#!
Babenko_Router_2(config)#ip dhcp pool poolvlan11
Babenko_Router_2(dhcp-config)# network 10.24.8.0 255.255.255.224
Babenko_Router_2(dhcp-config)# default-router 10.24.8.1
Babenko_Router_2(dhcp-config)# dns-server 10.24.9.140
Babenko_Router_2(dhcp-config)#ip dhcp pool poolvlan21
Babenko_Router_2(dhcp-config)# network 10.24.8.32 255.255.255.224
Babenko_Router_2(dhcp-config)# default-router 10.24.8.33
Babenko_Router_2(dhcp-config)# dns-server 10.24.9.140
Babenko_Router_2(dhcp-config)#ip dhcp pool poolvlan31
Babenko_Router_2(dhcp-config)# network 10.24.8.64 255.255.255.224
Babenko_Router_2(dhcp-config)# default-router 10.24.8.65
Babenko_Router_2(dhcp-config)# dns-server 10.24.9.140
Babenko_Router_2(dhcp-config)#ip dhcp pool poollan5
Babenko_Router_2(dhcp-config)# network 10.24.9.0 255.255.255.192
Babenko_Router_2(dhcp-config)# default-router 10.24.9.1
Babenko_Router_2(dhcp-config)# dns-server 10.24.9.140
```

Рисунок 3.21 – Налаштування DHCP для VLAN

Тепер, коли DHCP-пули успішно налаштовані, можемо перейти до налаштування віртуальних підінтерфейсів на маршрутизаторі (рис. 3.22).

```
Babenko_Router_2(config)#interface GigabitEthernet0/1.11
Babenko_Router_2(config-subif)# encapsulation dot1Q 11
Babenko_Router_2(config-subif)# ip address 10.24.8.1 255.255.255.224
Babenko_Router_2(config-subif)#!
Babenko_Router_2(config-subif)#interface GigabitEthernet0/1.21
Babenko_Router_2(config-subif)# encapsulation dot1Q 21
Babenko_Router_2(config-subif)# ip address 10.24.8.33 255.255.255.224
Babenko_Router_2(config-subif)#!
Babenko_Router_2(config-subif)#interface GigabitEthernet0/1.31
Babenko_Router_2(config-subif)# encapsulation dot1Q 31
Babenko_Router_2(config-subif)# ip address 10.24.8.65 255.255.255.224
Babenko_Router_2(config-subif)#!
Babenko_Router_2(config-subif)#interface GigabitEthernet0/1.99
Babenko_Router_2(config-subif)# encapsulation dot1Q 99
Babenko_Router_2(config-subif)# ip address 10.24.8.97 255.255.255.224
```

Рисунок 3.22 – Створення підінтерфейсів VLAN

Для забезпечення безпеки серверів, на порти комутаторів, що підключених до них, було налаштовано функцію безпеки портів, де ми дозволяємо доступ до порту тільки двом унікальним пристроям, налаштовуємо динамічне розпізнавання MAC-адрес та додавання їх в поточну конфігурацію, а також, щоб у разі порушення безпеки, з’являлось повідомлення, але доступ до порта залишався відкритим за допомогою наступних команд:

```
Babenko_Switch1_LAN5(config)#interface f0/24
Babenko_Switch1_LAN5(config-if)#switchport mode access
Babenko_Switch1_LAN5(config-if)#switchport port-security maximum 2
Babenko_Switch1_LAN5(config-if)#switchport port-security mac-address sticky
Babenko_Switch1_LAN5(config-if)#switchport port-security violation restrict
```

Рисунок 3.23 – Налаштування безпеки портів на Babenko_Switch1_LAN5

Для тестування роботи безпеки портів підключемо до f0/24 інший сервер з іншою MAC-адресою. Після відображення стану безпеки на рисунку рис. 3.24 ми бачимо, що конфігурація на `Fa0/24` дозволяє лише одну MAC-адресу вивчати на порту. Оскільки сталося порушення безпеки, порт наразі перебуває в обмеженому стані. Це означає, що нові MAC-адреси не вивчатимуться, і лише наразі вивчена MAC-адреса зможе обмінюватися даними через порт. Таким чином можемо зробити висновки, що функція безпеки портів налаштовано.

```
Babenko_Switch1_LAN5#sh port-security
Secure Port MaxSecureAddr CurrentAddr SecurityViolation Security Action
               (Count)       (Count)        (Count)
-----------------------------------------------------------------------
       Fa0/24        1          1                1         Restrict
-----------------------------------------------------------------------
```

Рисунок 3.24 – Тестування роботи безпеки портів

3.8 Перевірка роботи моделі комп’ютерної системи

Щоб перевірити базові налаштування маршрутизатора ми спробуємо здійснити спробу логіну до консолі (рис. 3.25):

```
Press RETURN to get started!

Attention! BabenkoOlA

User Access Verification

Username: 123201_Babenko
Password:
Babenko Router 3>
```

Рисунок 3.25 – Вхід до консолі Babenko_Router_3

Далі перевірка таблиці маршрутизації на граничному маршрутизаторі Babenko_Router_3, яка дає нам зрозуміти що налаштовані статичні маршрути працюють, а також у роботі протокол OSPF (рис. 3.26).

```
Gateway of last resort is 209.165.202.2 to network 0.0.0.0

     10.0.0.0/8 is variably subnetted, 15 subnets, 5 masks
O       10.0.1.0/30 [110/1562] via 10.0.1.18, 00:42:17, Serial0/2/0
O       10.0.1.4/30 [110/1562] via 10.0.1.18, 00:42:17, Serial0/2/0
O       10.0.1.8/30 [110/782] via 10.0.1.13, 01:12:22, Serial0/3/0
C       10.0.1.12/30 is directly connected, Serial0/3/0
L       10.0.1.14/32 is directly connected, Serial0/3/0
C       10.0.1.16/30 is directly connected, Serial0/2/0
L       10.0.1.17/32 is directly connected, Serial0/2/0
O       10.24.8.0/27 [110/782] via 10.0.1.18, 00:42:17, Serial0/2/0
O       10.24.8.32/27 [110/782] via 10.0.1.18, 00:42:17, Serial0/2/0
O       10.24.8.64/27 [110/782] via 10.0.1.18, 00:42:17, Serial0/2/0
O       10.24.8.96/27 [110/782] via 10.0.1.18, 00:42:17, Serial0/2/0
O       10.24.8.128/25 [110/783] via 10.0.1.13, 01:12:22, Serial0/3/0
O       10.24.9.0/26 [110/782] via 10.0.1.18, 00:42:17, Serial0/2/0
O       10.24.9.64/26 [110/783] via 209.165.202.2, 01:12:22, Serial0/3/1
O       10.24.9.128/27 [110/782] via 10.0.1.13, 01:12:53, Serial0/3/0
     64.0.0.0/30 is subnetted, 1 subnets
O       64.100.13.0/30 [110/782] via 209.165.202.2, 01:12:22, Serial0/3/1
     209.165.200.0/27 is subnetted, 1 subnets
S       209.165.200.32/27 [1/0] via 209.165.202.2
     209.165.201.0/28 is subnetted, 1 subnets
S       209.165.201.0/28 [1/0] via 209.165.202.2
     209.165.202.0/24 is variably subnetted, 2 subnets, 2 masks
C       209.165.202.0/30 is directly connected, Serial0/3/1
L       209.165.202.1/32 is directly connected, Serial0/3/1
S*   0.0.0.0/0 [1/0] via 209.165.202.2

Babenko_Router_3#
```

Рисунок 3.26 – Таблиця маршрутизації Router3

Наступною перевіркою є можливість пристроїв мережі виходити в Інтернет, зробимо спробу пінга з однієї з мереж на комп'ютер що під'єднано до ISP, що не є частиною корпоративної мережі, результат на рисунку 3.27.

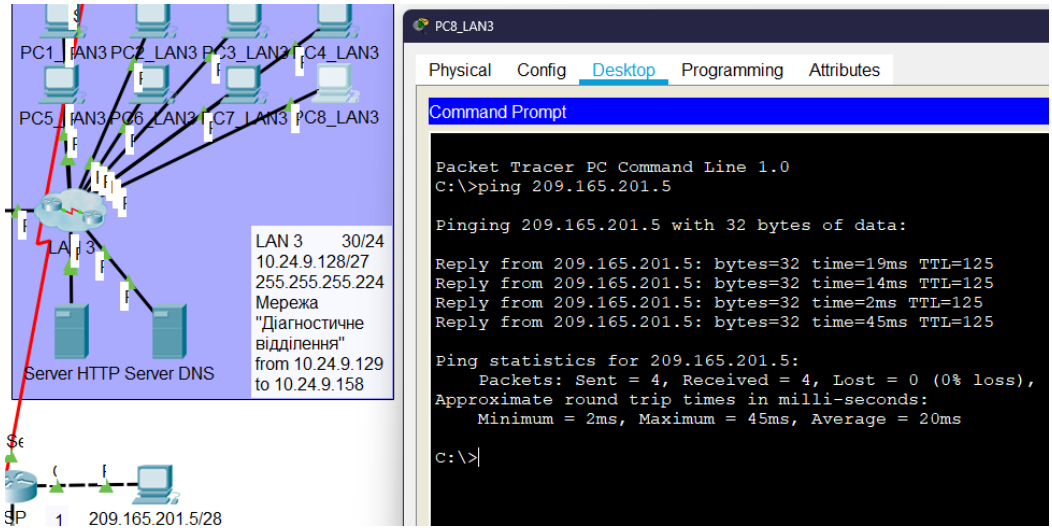

Рисунок 3.27 – Пінг з мережі до Інтернету

Далі перевіряємо працездатність NAT, чи працює підміна адреси при виході пакету з мережі до Інтернету. Робимо пінг з хосту в мережі на адресу

комп’ютера, що під’єднано до ISP, заходимо на маршрутизатор і за допомогою команди дивимось трансляцію адреси (рис. 3.28).

```
Babenko_Router_3(config)#do show ip nat translation
Pro  Inside global     Inside local       Outside local      Outside global
icmp 209.165.200.5:7   10.24.9.141:7      209.165.201.5:7    209.165.201.5:7
```

Рисунок 3.28 – Трансляція NAT на Router3

Повторюємо ті самі дії, але вже для іншого граничного маршрутизатора, отримуємо результат на рисунку 3.29.

```
Babenko_Router_5(config)#do show ip nat translation
Pro  Inside global     Inside local       Outside local      Outside global
icmp 209.165.200.37:1  10.24.9.82:1       209.165.201.5:1    209.165.201.5:1
```

Рисунок 3.29 – Трансляція NAT на Router5

Також, з рисунку нижче можемо побачити що йде трансляція IP-адреси 209.165.200.4 на 10.24.9.139 що є IP-адресою HTTP-серверу (рис. 3.30).

```
Babenko_Router_3(config)#do show ip nat translation
Pro  Inside global     Inside local       Outside local      Outside global
icmp 209.165.200.5:7   10.24.9.141:7      209.165.201.5:7    209.165.201.5:7
---  209.165.200.4     10.24.9.139        ---                ---

Babenko Router 3(config)#
```

```
Pro  Inside global     Inside local       Outside local      Outside global
icmp 209.165.200.37:1  10.24.9.82:1       209.165.201.5:1    209.165.201.5:1
icmp 209.165.200.37:2  10.24.9.82:2       209.165.201.5:2    209.165.201.5:2
---  209.165.200.4     10.24.9.139        ---                ---

Babenko Router 5(config)#
```

Рисунок 3.30 – Трансляція NAT на HTTP сервер

Зробимо спробу запиту на адресу 209.165.200.4 з одного з комп’ютерів віддаленої мережі MED1 (рис. 3.31).

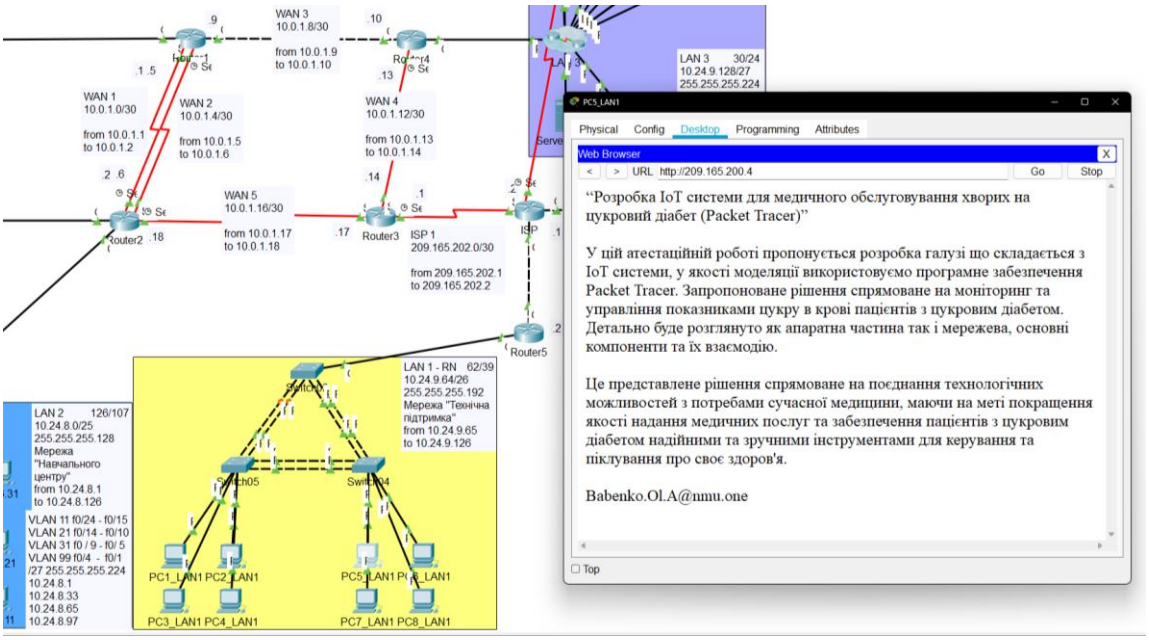

Рисунок 3.31 – Перевірка доступності HTTP-серверу за public-адресою

Перевіримо доступність HTTP та DNS серверів за допомогою спроби запиту з комп'ютера Інтернет-провайдера на 123.dnipro.ua, результат на рисунку 3.32.

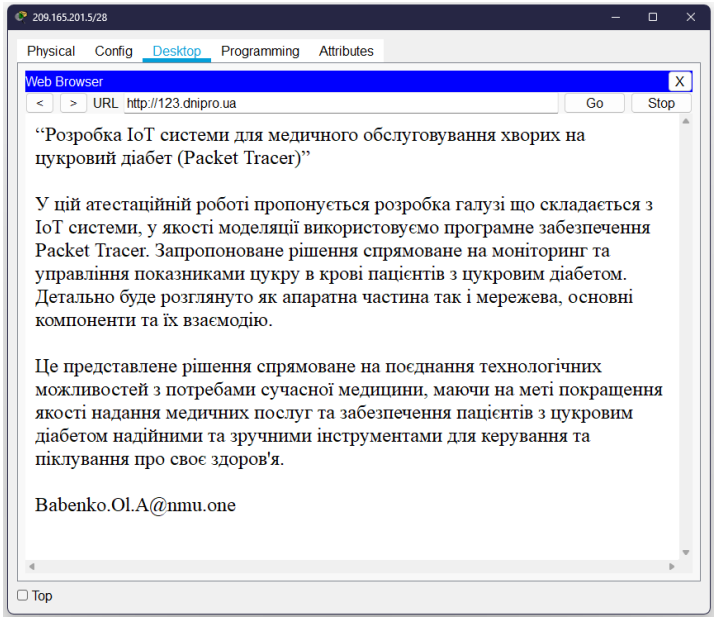

Рисунок 3.32 – Перевірка доступності адреси 123.dnipro.ua

Для перевірки VPN-з'єднання виконаємо перевірку наявності підключення за допомогою команди show crypto isakmp sa на пристрої Router3 (рис. 3.33).

```
Babenko_Router_3#show crypto ?
  ipsec   Show IPSEC policy
  isakmp  Show ISAKMP
  key     Show long term public keys
  map     Crypto maps
Babenko_Router_3#show crypto key ?
  mypubkey  Show public keys associated with this router
Babenko_Router_3#show crypto isakmp ?
  policy  Show ISAKMP protection suite policy
  sa      Show ISAKMP Security Associations
Babenko_Router_3#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
64.100.13.2     209.165.202.1   QM_IDLE           1064    0 ACTIVE


IPv6 Crypto ISAKMP SA


Babenko_Router_3#
```

Рисунок 3.33 – Перевірка існування VPN з'вязку

Тепер впевнимося що пінг з віддаленої мережі до корпоративної мережі йде успішно, робимо запит з підмережі де раніше налаштовували VLAN (рис. 3.34).

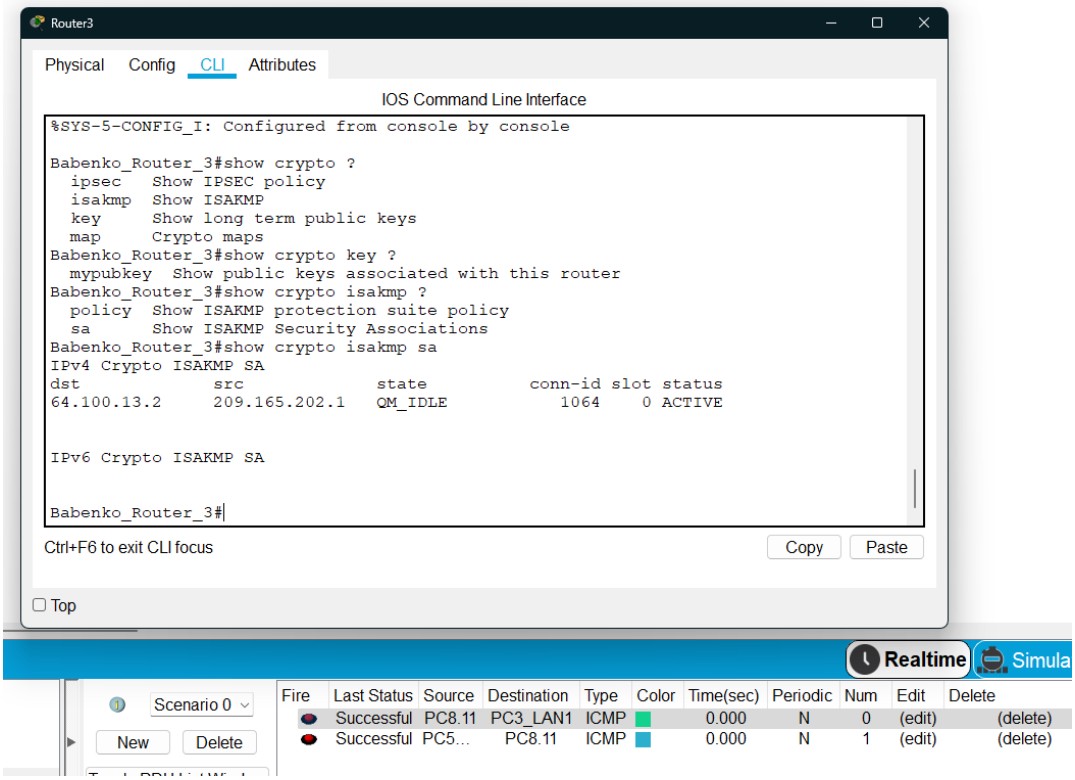

Рисунок 3.34 – Спроба пінгу завдяки застосуванню VPN-з’єднання

Останнім кроком перевірки є впевнитись що налаштування VLAN було зроблено вірно та пристрої з різних VLAN’ів можуть комунікувати між собою, для цього на одному з пристроїв, що знаходиться у VLAN 11 зробимо запит на пристрій, який знаходиться у VLAN 21.

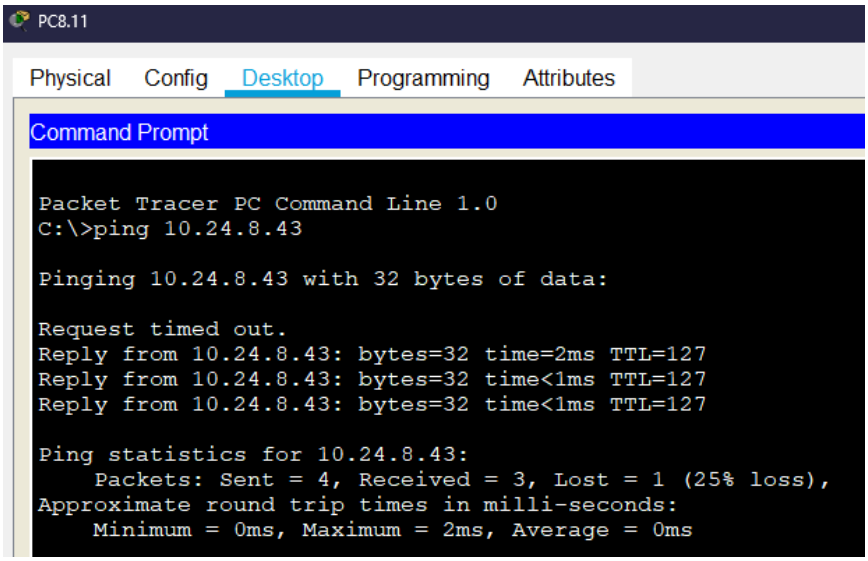

Рисунок 3.35 – Перевірка працездатності VLAN

4 РОЗРОБКА IoT СИСТЕМИ МЕДИЧНОГО ОБСЛУГОВУВАННЯ ХВОРИХ НА ЦУКРОВИЙ ДІАБЕТ

4.1 Розробка архітектури IoT-системи МОХЦД

В проєкті пропонується рішення для моніторингу здоров'я пацієнта з діабетом. IoT-система МОХЦД має пристрої, які в повсякденному житті пацієнт постійно носить для моніторингу рівня глюкози, і фітнес-трекер для моніторингу рівня фізичного навантаження та дихання. Ці пристрої обробляють а надсилають дані в заклад, що займається моніторингом здоров’я, щоб вжити відповідних заходів, коли стан здоров’я пацієнта виходить за межі нормального діапазону.

На рисунку 4.1 схема алгоритма рішення IoE для моніторингу пацієнта з діабетом.

Запропонована система складається з трьох основних компонентів: підсистема моніторингу стану здоров’я, підсистема обробки та підсистема телемедицини.

Підсистема моніторингу стану здоров’я пацієнта включає в себе носимі пристрої, які виконують постійний контроль рівня глюкози в крові, а також відстеження інших важливих показників таких як артеріальний тиск, частота серцевих скорочень та рівень фізичної активності через фітнес-трекер. Ця інформація постійно надсилається на мобільний пристрій через з’єднання Bluetooth. Мобільний пристрій зі встановленим клієнським додатком надсилає цю інформацію до підсистеми обробки і зберігання в режимі реального часу через мережу 4G або Wi-Fi.

Підсистема обробки та зберігання даних робить аналіз і обробку даних для виявлення тенденцій та аномалій у стані здоров’я пацієнта. Він складається з мобільного додатку та сервера бази даних. Сервер бази даних може бути локальним або хмарним та збирає дані з датчиків. Згодом ці дані аналізуються, щоб знайти тенденції, які використовуються для визначення, чи може пацієнт потребувати негайної допомоги.

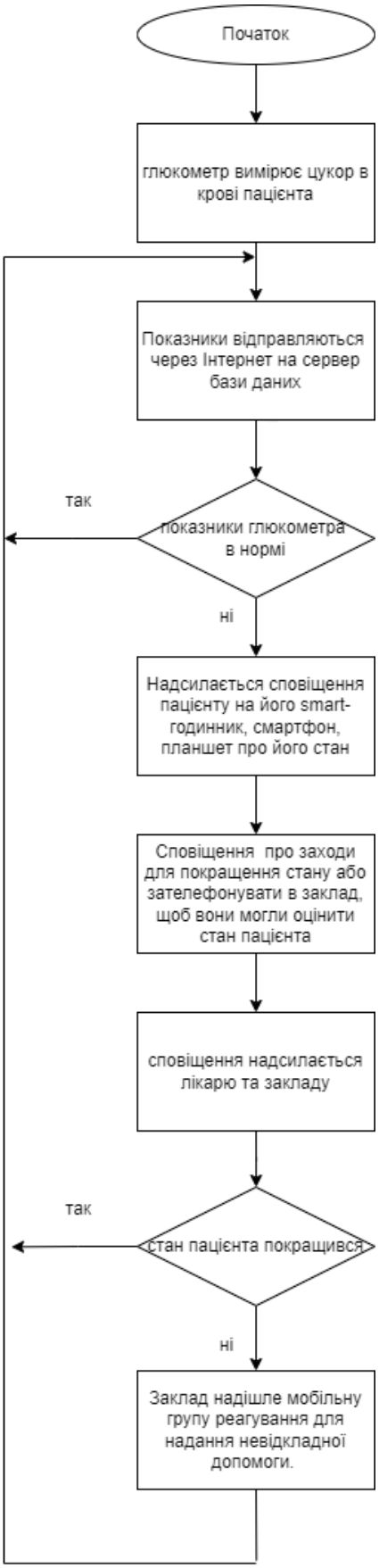

Рисунок 4.1 – Схема алгоритма рішення IoE для моніторингу пацієнта з діабетом

Підсистема телемедицини має на меті забезпечення дистанційного спостереження за станом здоров'я пацієнта та консультацій з лікарем, а також надання швидкої медичної допомоги в екстрених ситуаціях. Складається з мобільного додатку лікаря та системи моніторингу. Система моніторингу аналізує дані, отримані від датчиків. Коли система виявляє нестандартну ситуацію, сповіщення надсилається лікарю, щоб на мобільному додатку лікаря визначити, чи це надзвичайна ситуація та відправить МГР для надання невідкладної допомоги пацієнту, чи просто порушення використання ліків пацієнтом із діабетом.

На рисунку 4.2 показана розроблена архітектура системи моніторингу діабету.

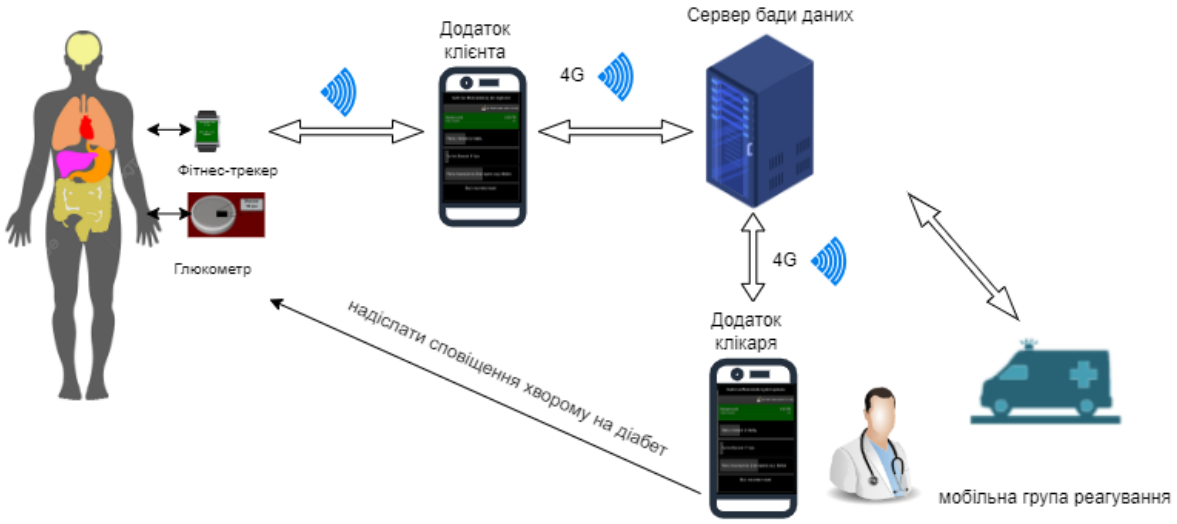

Рисунок 4.2 – Архітектура IoT-МОХЦД

4.2 Розробка моделі IoT-системи в Packet Tracer

Для моделювання IoT системи медичного обслуговування хворих на цукровий діабет та мережі медичних лікарень було прийнято рішення використовувати програму для моделювання IoT-речей: Cisco Packet Tracer.

На рисунку 4.3 змодельована модель в Packet Tracer, яка складається з 4-х кластерів: будинок пацієнта, мобільна група реагування, лікарня та IoT-сервер.

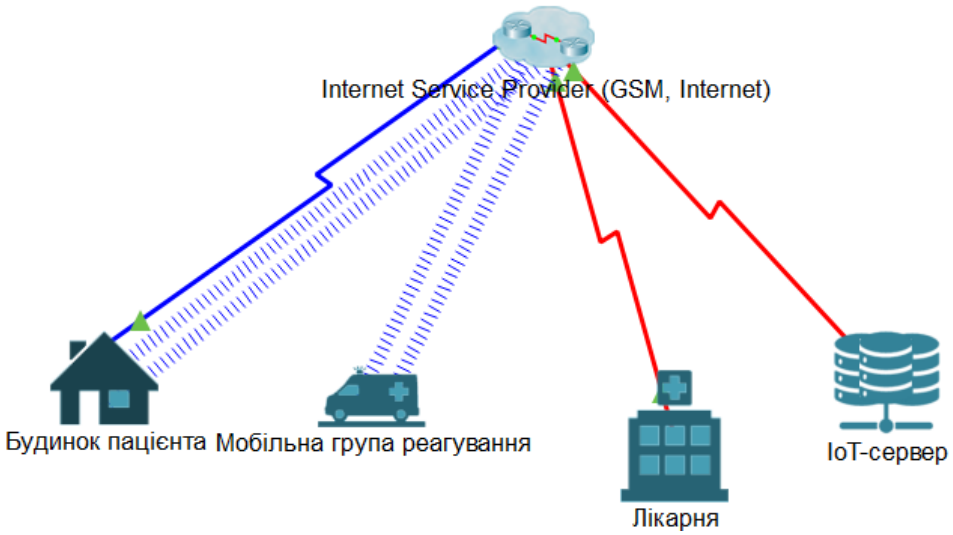

Рисунок 4.3 – Модель IoT-системи МОХЦД

В кластері «Будинок пацієнта» (рис. 4.4) додані IoT-пристрої:

– фітнес-трекер, щоб стежити за рівнем фізичних навантажень і частотою дихання,;

– імплантований безперервний глюкометр, який вимірює та обробляє рівень глюкози пацієнта.

Ці пристрої встановлюються вдома у пацієнта. IoT-пристрої використовують технологію зв'язку Wi-Fi для створення бездротового підключення до Home Gateway. Home Gateway через кабельний модем під’єднаний до мережі постачальника Інтернет-послуг (Internet Service Provider, ISP). Ця інформація постійно надсилається на мобільні пристрої ї (смартфон та планшет) зі встановленим клієнтським додатком через з’єднання Bluetooth. Мобільні пристрої надсилають цю інформацію до підсистеми обробки і зберігання в режимі реального часу через мережу 4G.

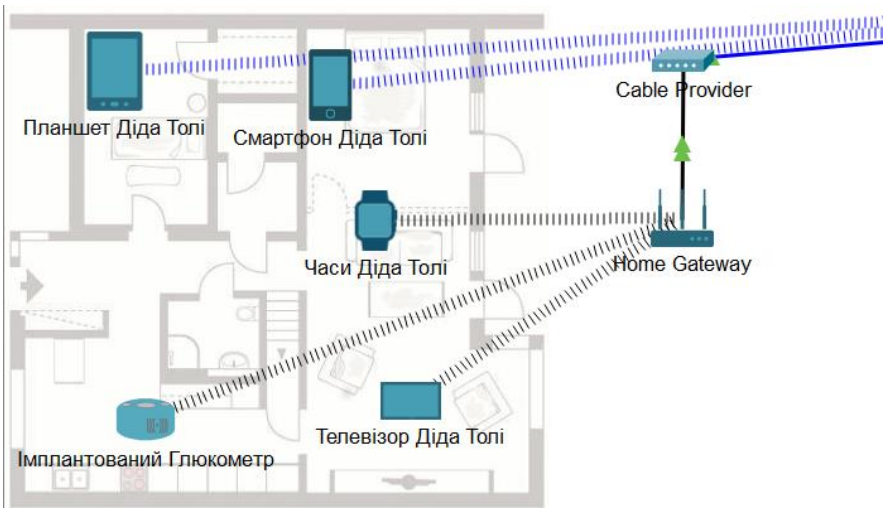

Рисунок 4.4 – Кластер «Будинок пацієнта»

До ISP належать бездротові та мобільні мережі 3G, які з'єднують пристрої моніторингу та звітності пацієнтів (розумний годинник, вимірювач рівня глюкози, смартфон, смарт-телевізор, планшет тощо) із хмарним IoT-сервером, лікарнею та мобільною групою реагування (МГР).

Кластер «IoT-сервер» для зберігання медичних хмарних даних. Зазвичай зберігаються у віддалених хмарних центрах обробки даних, які мають забезпечувати безпечне зберігання (рис. 4.5).

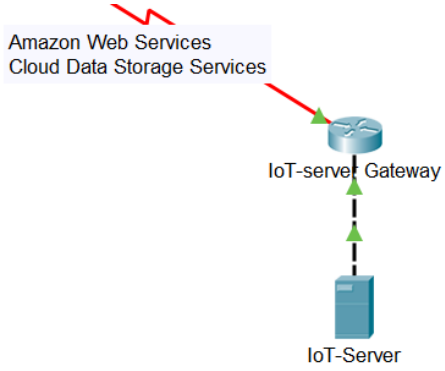

Рисунок 4.5 – Кластер «IoT-сервер»

Кластер «Лікарня» представляє мережу медичного закладу, розроблену в розділі 3. Сюди входить 5 розроблених підмерж MED1-MED5.

Кластер «Мобільна група реагування» (рис. 4.6) предназначена для швидкого реагування у разі погіршення стану пацієнта для надання йому невідкладної медичної допомоги. Представлена двома планшетами з підтримкою

4G зі встановленим програмним забезпеченням для лікарів для зв’язку з IoT-сервером.

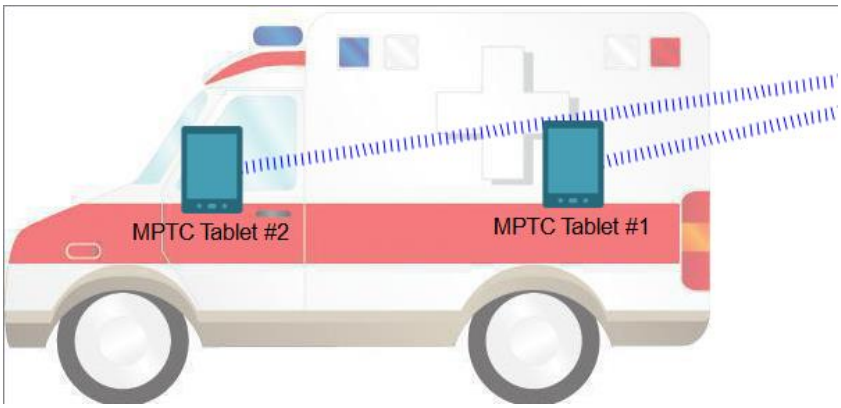

Рисунок 4.6 – Кластер «Мобільна група реагування»

4.3 Розробка Script у Cisco Packet Tracer для моделювання IoT-системи

Cisco Packet Tracer – це потужний інструмент для мережевого моделювання, що дозволяє студентам та фахівцям створювати, налаштовувати та тестувати мережі. Однією з ключових особливостей Packet Tracer є можливість використовувати скрипти Python для автоматизації завдань та створення складних сценаріїв.

Модуль PT Script у Cisco Packet Tracer – це потужний інструмент, який дозволяє автоматизувати завдання, створювати користувальницьку поведінку та розширювати функціональні можливості симуляцій. Він використовує сценарії Python, що робить його доступним для широкого кола користувачів.

Вкладка «Custom Interfaces» використовується для додавання, видалення, редагування, перейменування, імпорту та експорту файлів спеціального інтерфейсу. Спеціальні інтерфейси закодовані у html (рис. 4.7).

Були розроблені інтерфейси для IoT-пристроїв в кластері «Будинок пацієнта», а саме для глюкометра (glicose_meter.htm) та фітнес-браслета (watch.htm), а також інтерфейси застосунків для додатку клієнта (patient_view.htm), додатку лікарів (ambulance_view.htm), інформаційної панелі контролера стану, щоб моделювати різні стани пацієнта.

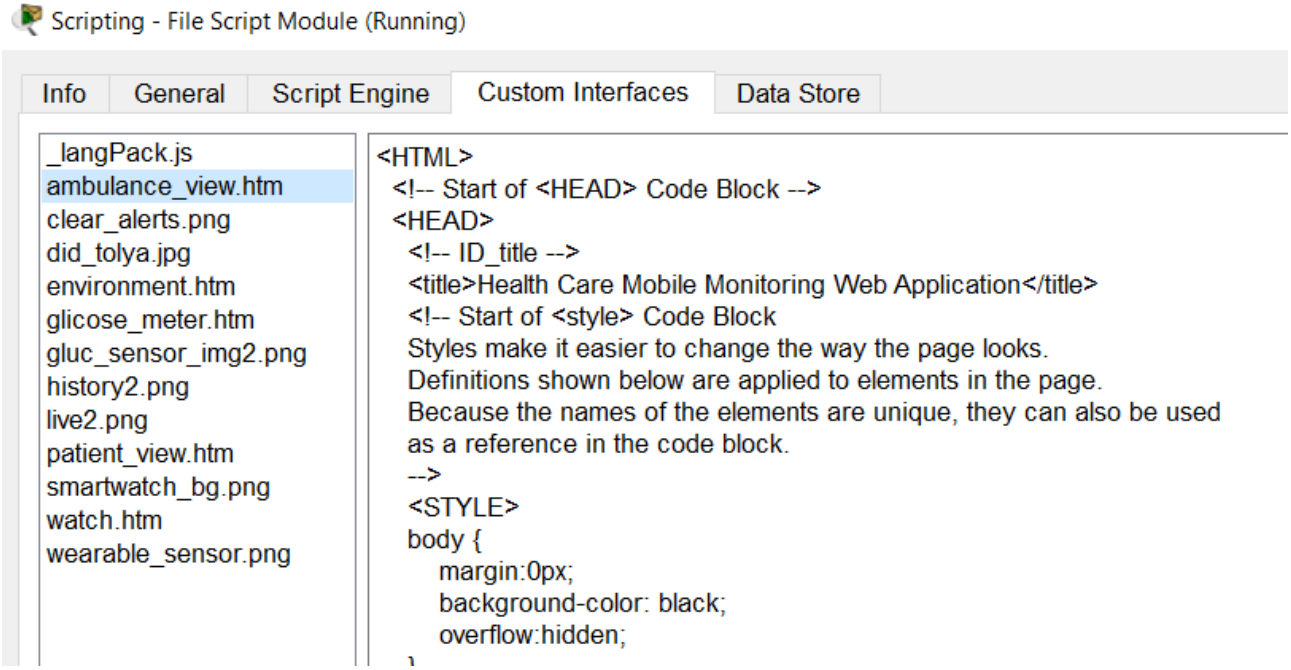

Рисунок 4.7 – Custom Interfaces

Створені інтерфейси в вікні «Configure Custom Interfaces» обиралися для відповідного пристрою (рис.4.8).

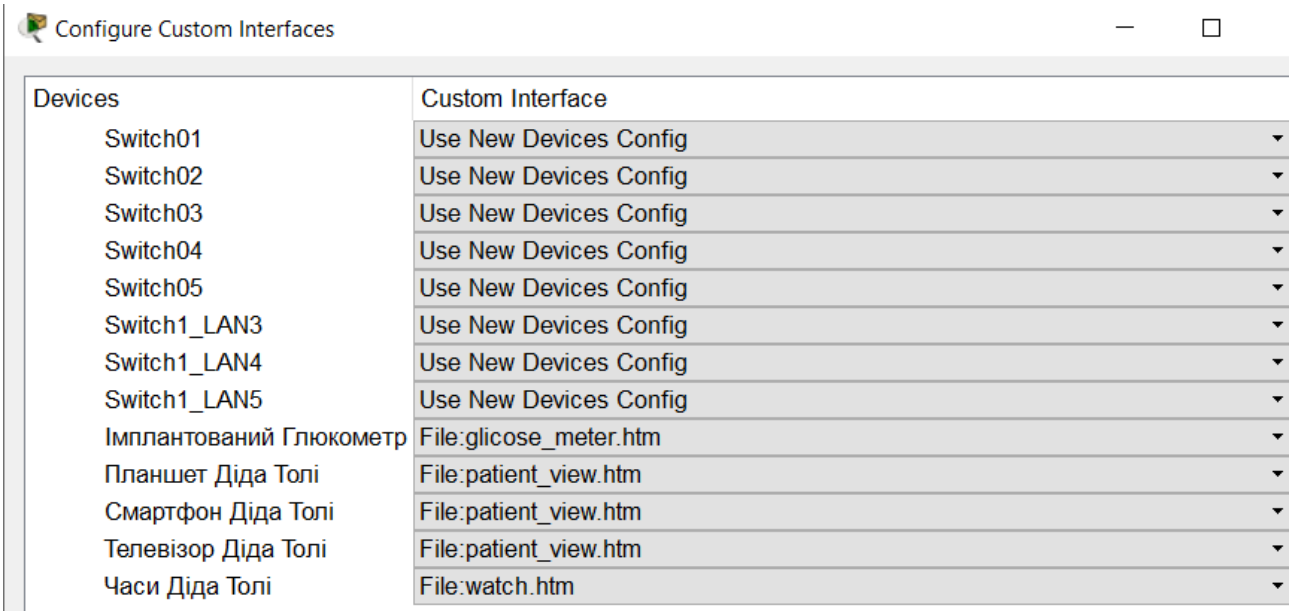

Рисунок 4.8 – Configure Custom Interfaces

В результаті, налаштовані пристрої матимуть нову вкладку, наприклад для смартфона на рисунку 4.9. Ця вкладка демонструє додаток для пацієнта. В додатку відображається поточний стан показників рівня глюкози та фізичного стану, а в випадку погіршення показників сповіщення з рекомендаціями від медичного закладу та доктора.

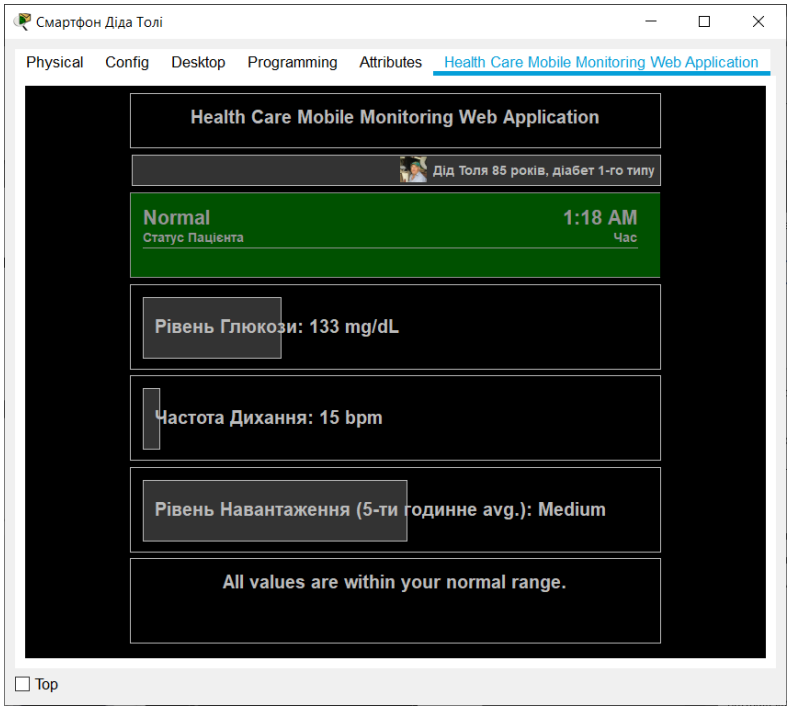

Рисунок 4.9 – Вкладка з створеним інтерфейсом для смартфона

4.4 Розробка інтерфейсу мобільного застосунку для пацієнта

Розробка інтерфейсу мобільного застосунку (рис. 4.9) велась за допомогою мови HTML, а відладка засобами Cisco Packet Tracer. За допомогою стандартних засобів роботи мови HTML, було створено блоки, що описують вигляд елементів. Нижче наведено приклад описування фону застосунку, що було застосовано у програмі:

```
#app {
        width:100%;
        height:100%;
        top:0px;
        margin:0px;
        color: #bbbbbb;
        background-image:url('bg1.png');
        background-repeat:no-repeat;
        background-position:right bottom;
        background-color: black;
        border: 1px solid black;
        margin-left:auto;
        margin-right:auto;
}
```

Потім ми створили блок оголошених змінних, що потрібні нам для роботи функцій застосунку (рис. 4.10).

```
// Початок блоку оголошення змінних
var someEvent = 0;
var speedUpTime = 5000;
var currCond = 2;
var latest = null;
var date = new Date();
var trackerID;
var INTERVAL = 5000;
var initWidth = 480;
var initHeight = 480;
var critical = false;
var dispatched = false;
//var ETA = 0;
var refresh = true;
var doorLocked = true;
var ambulanceArrived=false;
// defining the normal ranges
var glucFlo = 115;
var glucCei = 200;
var respFlo = 70;
var respCei = 80;
var exercFlo = 98;
var exercCei = 99.2;
var glucHi = glucLo = exercHi = exercLo = respHi = respLo = 0;
var lp;
// Кінець блоку оголошення змінних
```

Рисунок 4.10 – Блок оголошення змінних застосунку

Переходимо до створення функцій, створили блок SCRIPT, де описали усі функції виконувані програмою, нижче наведено приклад однієї з написаних фукнцій, що виконує оновлення статусу стану здоров'я пацієнта (рис. 4.11).

```
function updateStatus() {
  //refresh = true;
  if (latest.condition == 0) {
    lowAlert();
    } else {
      if (latest.condition == 1) {
        hiAlert();
      } else {
        if (latest.condition == 2) {
          backToNormal();
        }
      }
    }
}
```

Рисунок 4.11 – Функція що описує вигляд програми при високому рівні цукру

Один з останніх етапів розробки інтерфейсу додатку для пацієнта було створення блоку BODY, де було описано відображення веб-сторінки, що є інтерфейсом користувача. Текст програми застосунку наведено в Додатку А.

Подібним чином було описано і вигляд інтерфейсу, що призначений для медичного персоналу, але з іншими значеннями змінних та трішки іншим блоком функцій, але у цілому, структура та сама.

4.5 Моделювання роботи IoT-системи МОХЦД

Розглянемо три випадки: нормальний, гіпоглікемія та гіперглікемія.

Нормальний випадок. У цьому випадку рівень глюкози знаходиться в межах норми. Пацієнт може контролювати свій рівень глюкози, рівень фізичних навантажень і частоту дихання за допомогою будь-якого зі своїх пристроїв, таких як смартфон (рис. 4.12). Показники передаються на IoT-сервер.

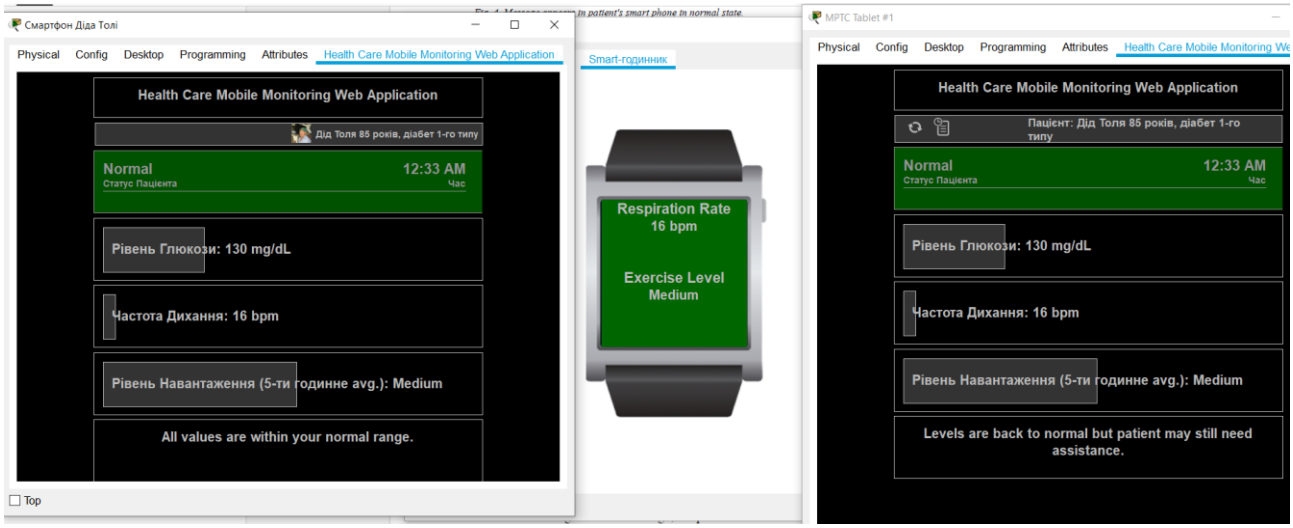

Рисунок 4.12 – Показники в межах норми

Гіпоглікемії. Рівень глюкози низький, пацієнт повинен приймати все, що містить цукор. На пристрої пацієнта який надсилаються попередження, і якщо пацієнт не реагує та рівень глюкози не підвищується, прибуде МГР, щоб надати допомогу пацієнту, як показано на рисунку 4.13.

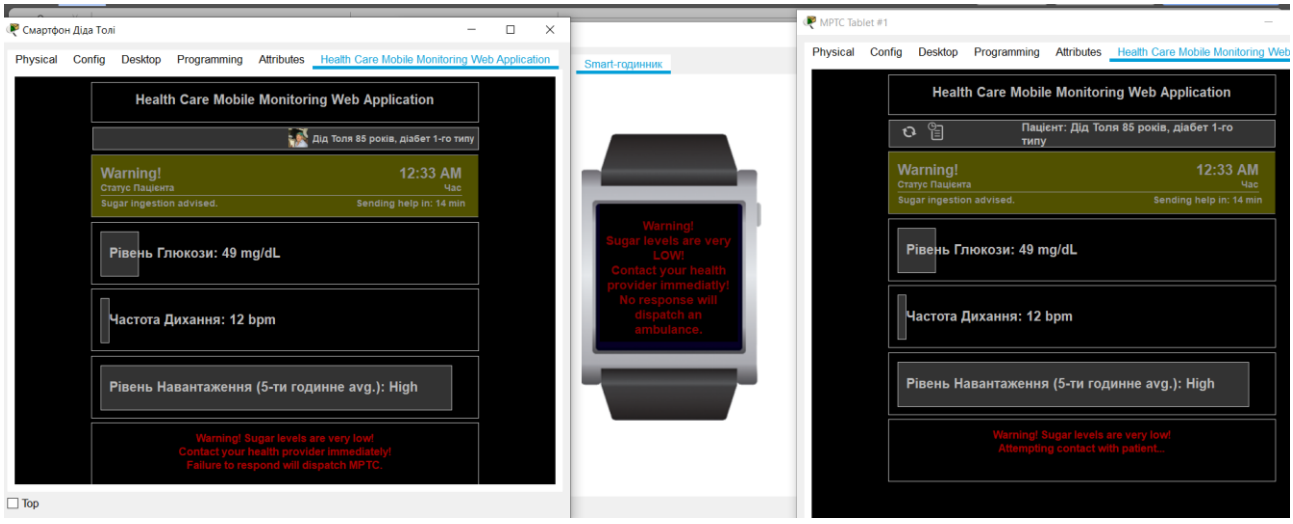

Рисунок 4.13 – Моделювання режиму гіпоглікемії

Гіперглікемія. У цьому випадку високий рівень глюкози, пацієнт повинен приймати інсулін. Безперервний глюкометр надсилає дані до МОХЦД, який надсилає попереджувальне повідомлення на пристрої пацієнта, і якщо пацієнт не відповідає, а рівень глюкози залишається в небезпечному діапазоні, МОХЦД надішле МГР, як показано на рис. 4.14.

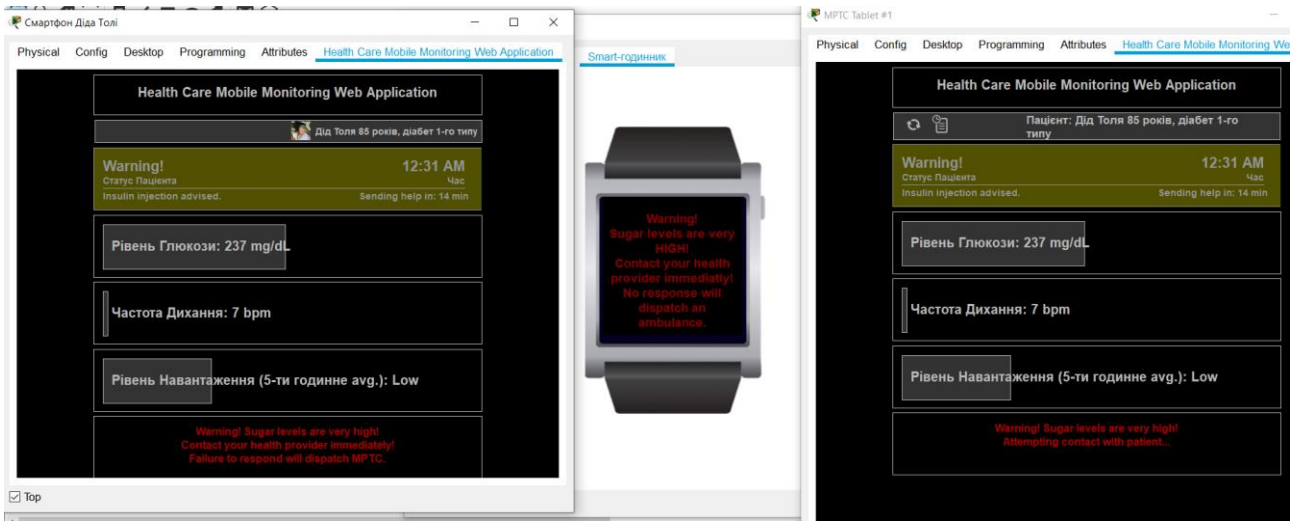

Рисунок 4.14 – Моделювання режиму гіперглікемії

ВИСНОВКИ

Отже, можемо зробити висновок що проектування комп’ютерної системи для центру піклування про здоров’я пацієнтів із захворюванням на цукровий діабет є достатньо важливим і принциповим моментом, що допомагає покращити життя та здоров’я пацієнтів, а також їх досвід комунікації з лікарнею та медичним персоналом.

Розроблена IoT-система була симульована за допомогою Packet Tracer, демонструючи її потенціал для надання швидкої та ефективної допомоги пацієнтам. В Packet Tracer були розроблені скрипти для моделювання роботи імплантованого глюкометра для моніторингу рівня глюкози в крові пацієнта, фітнес-трекера для моніторингу фізичного стану пацієнта, та мобільні додатки для пацієнта та лікаря.

Система дозволяє пацієнтові контролювати свій стан здоров'я, попереджаючи його про потенційні проблеми з рівнем глюкози.

Вона також може автоматично викликати медичну допомогу, якщо стан пацієнта загрожує його життю.

Розроблена система спостереження за станом здоров’я, націлена на допомогу хворим на цукровий діабет реагувати на їхню ситуацію та самостійно контролювати свою хворобу.

СПИСОК ДЖЕРЕЛ ПОСИЛАННЯ

1 Що треба знати про цукровий діабет: типи, симптоми, ускладнення | Центр громадського здоров'я. *Центр громадського здоров'я України | МОЗ*. URL: https://www.phc.org.ua/news/scho-treba-znati-pro-cukroviy-diabet-tipi-simptomi-uskladnennya (дата звернення: 04.06.2024).

2 Комп'ютерні технології в оптимізації лікарського забезпечення хворих на цукровий діабет | Інтернет-видання "Новини медицини та фармації". *Новости медицины в Украине | Інтернет-видання "Новини медицини та фармації"*. URL: http://urgent.mif-ua.com/archive/article/41620 (дата звернення: 04.06.2024)

3 Whiting, IDF diabetes atlas: Global estimates of the prevalence of diabetes for 2011 and 2030, Diabetes Res. Clin. Pract., № 94, с. 311, https://doi.org/10.1016/j.diabres.2011.10.029

4 IOE solution for a diabetic patient monitoring, SF Ismail, 2017 8th international conference on information technology (ICIT), 244-248

5 A Smart Glucose Monitoring System for Diabetic Patient / A. Rghioui та ін. *Electronics*. 2020. Т. 9, № 4. С. 678. URL: https://doi.org/10.3390/electronics9040678 (дата звернення: 01.07.2024).

6 Siddiqui, Pain-free blood glucose monitoring using wearable sensors: Recent advancements and future prospects, IEEE Rev. Biomed. Eng., № 11, с.

7 Amine Rghioui , Assia Naja, Jaime Lloret Mauri, Abedlmajid Oumnad. An IoT Based diabetic patient Monitoring System Using Machine Learning and Node MCU, The International Conference on Mathematics & Data Science (ICMDS) 2020, Journal of Physics: Conference Series 1743 (2021) 012035 IOP Publishing doi:10.1088/1742-6596/1743/1/012035

8 Методичні рекомендації до виконання кваліфікаційної роботи бакалавра студентами галузі знань 12 Інформаційні технології спеціальності 123 Комп'ютерна інженерія / Л.І. Цвіркун, С.М. Ткаченко, Я.В. Панферова, Д.О. Бешта, Л.В. Бешта. – Д.: НТУ «ДП», 2023. – 62 с.

9 ДСТУ 3008-2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. – К.: Держстандарт, 2015. – 37с.

10 Положення про організацію атестації здобувачів вищої освіти НТУ «Дніпровська політехніка» / М-во освіти і науки України, Нац. техн. ун-т. – Д. : НТУ «ДП», 2018. – 40 с 3

Додаток А

Текст програми мобільного застосунку пацієнта

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ДНІПРОВСЬКА ПОЛІТЕХНІКА”

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
МОБІЛЬНИЙ ЗАСТОСУНОК ПАЦІЄНТА

Текст програми

804.02070743.23004-01 12 01

Листів 12

2024

АНОТАЦІЯ

Дана програма містить в собі програмний код мобільного застосунку Health Care для IoT-системи медичного обслуговування хворих на цукровий діабет.

Програма призначена для забезпечення моніторингу та контролю за станом здоров’я пацієнта, відправка мобільної групи регування у разі потреби, а також інформування пацієнта щодо показників стану його здоров’я.

Програма написана мовою JavaScript з використанням мови HTML, відлагоджена із застосуванням середовища Cisco Packet Tracer.

```
<HTML>
  <!-- Start of <HEAD> Code Block -->
  <HEAD>
      <!-- ID_title -->
      <title>Health Care Mobile Monitoring Web Application</title>
      <!-- Start of <style> Code Block
      Styles make it easier to change the way the page looks.
      Definitions shown below are applied to elements in the page.
      Because the names of the elements are unique, they can also be used
      as a reference in the code block.
      -->
    <STYLE>
      body {
            margin:0px;
            background-color: black;
            overflow:hidden;
      }
      h1 {

            position: relative;
            font-size: 20px;
            font-weight: bold;
            font-family: 'Roboto', sans-serif;
            margin-left:auto;
            margin-right:auto;
        }
      h2 {

            position: relative;
            font-size: 17px;
            font-family: 'Roboto', sans-serif;
            font-weight: bold;
            margin-left:auto;
            margin-right:auto;

      }
      .button {
            border: 1px solid #777777;
            float: right;
            color:#bbbbbb;
            font-size: 11px;
            font-family: 'Roboto', sans-serif;
            text-decoration:none;
            background: #660000;
      }
      .text {
            width: 88%;
            height:50px;
            position: relative;
            font-size: 15px;
            font-weight: bold;
            font-family: 'Roboto', sans-serif;
            margin-left:auto;
            margin-right:auto;
            margin-top: 5px;
            margin-bottom: 5px;
            padding: 10px;
            border: 1px solid #bbbbbb;
      }
      .alerts {
            width: 88%;
```

```
      height:50px;
      position: relative;
      font-size: 12px;
      font-weight: bold;
      font-family: 'Roboto', sans-serif;
      color: #aa0000;
      margin-left:auto;
      margin-right:auto;
      margin-top: 5px;
      margin-bottom: 5px;
      padding: 10px;
      border: 1px solid #bbbbbb;
}
.labels {
      position: relative;
      top: -51px;
      font-size: 15px;
      font-weight: bold;
      font-family: 'Roboto', sans-serif;
      margin-top: 5px;
      margin-bottom: 5px;
      padding: 10px;

}

</STYLE>

<SCRIPT>
   var xml_lang = "media_3.3.2.2.xml";
   var langPack = [];

   function Entry() {
     this.id ='' ;
     this.text = '';
   }

   function addEntry(id,text) {
     var nEntry = new Entry;
     nEntry.id = id;
     nEntry.text = text;
     langPack.push(nEntry);
     return false;
   }

   function fetchXML(f) {
      if (window.XMLHttpRequest)
       {
         var xhttp=new XMLHttpRequest();
       }
       else // for IE 5/6
        {
         var xhttp=new ActiveXObject("Microsoft.XMLHTTP");
        }
      xhttp.open("GET",f,false);
      xhttp.send();
      var xmlDoc=xhttp.responseXML;
      prepare_xmlObj(xmlDoc);
      return;
   }

   function prepare_xmlObj(o) {
         var l = o.getElementsByTagName('component').length;
         var nId, nTxt = "";
```

```
            for (var i=0;i<l;i++) {
               nId = o.getElementsByTagName('component')[i].getAttribute('id');
               nTxt= o.children[0].children[i].children[0].innerHTML;
               nTxt= nTxt.replace("<![CDATA[", "").replace("]]>", "");
               addEntry(nId,nTxt);
            }
            console.log(langPack[43]);
            console.log(langPack[44]);
         }

         function applyLangPack(lang) {
            lp = JSON.parse(lang);
            dprint("running applylangPack on ambulance_view. langPack =
"+lp.ID_0);
          document.getElementById("ID_txt21").innerHTML = lp.ID_21;
            document.getElementById("username").innerHTML = lp.ID_22;
            document.getElementById("statusValue").innerHTML = lp.ID_23;
            document.getElementById("ID_txt24").innerHTML = lp.ID_24;
            document.getElementById("ID_txt25").innerHTML = lp.ID_25;
            document.getElementById("recommText").innerHTML = lp.ID_26;
            document.getElementById("warningETA").innerHTML = lp.ID_27;
            document.getElementById("ambulanceText").innerHTML = lp.ID_28;
            document.getElementById("ambulanceValue").innerHTML = lp.ID_29;
            document.getElementById("ID_txt30").innerHTML = lp.ID_30;
            document.getElementById("glucValue").innerHTML = lp.ID_31;
            document.getElementById("ID_txt32").innerHTML = lp.ID_32;
            document.getElementById("respValue").innerHTML = lp.ID_33;
            document.getElementById("ID_txt34").innerHTML = lp.ID_34;
            document.getElementById("exercValue").innerHTML = lp.ID_35;
            document.getElementById("alertTxt").innerHTML = lp.ID_59;
            document.getElementById("ID_txt60").innerHTML = lp.ID_60;
            document.getElementById("hypoHistory").innerHTML = lp.ID_61;
            document.getElementById("hyperHistory").innerHTML = lp.ID_62;
            document.getElementById("normalHistory").innerHTML = lp.ID_63;
         }
           // ---------------------- End of the translation support functions
block ----------------------------------


         // Start of variable declaration block
         var someEvent = 0;
         var speedUpTime = 5000;
         var currCond = 2;
         var latest = null;
         var date = new Date();
         var trackerID;
         var INTERVAL = 5000;
         var initWidth = 480;
         var initHeight = 480;
         var critical = false;
         var dispatched = false;
         //var ETA = 0;
         var refresh = true;
         var doorLocked = true;
         var ambulanceArrived=false;
         var showingLogs = false;
         // defining the normal ranges
         var glucFlo = 115;
         var glucCei = 200;
         var respFlo = 70;
         var respCei = 80;
         var exercFlo = 98;
```

```
         var exercCei = 99.2;
         var glucHi = glucLo = exercHi = exercLo = respHi = respLo = 0;
         var lp;
         // End of variable declaration block


         function init() {
             //dprint(webView.getWebViewId());
             //dprint(webView.getObjectUuid());

             if       (someEvent       ==       1)       $se("subscribeWebviews",
webView.getWebViewId());
              $se("clientSubscriber", webView.getWebViewId());
            document.onkeyup = captureKey;
            show();
            document.getElementById("app").style.width = initWidth;
            document.getElementById("app").style.height = initHeight;
             $se("applyLangPackToWins", webView.getWebViewId());
            return;
         }

         function captureKey(e){
           var unicode=e.keyCode? e.keyCode : e.charCode
           actualkey=String.fromCharCode(unicode)
           switch (actualkey) //captura teclas normais
             {
               case "C":
               toggleCritical();
               break;
               }
           }

      function callbackFunc(v) {
            dprint("in callback now");
            latest = JSON.parse(v);
      }
         function show(){
            var Digital=new Date()
            var hours=Digital.getHours()
            var minutes=Digital.getMinutes()
            var seconds=Digital.getSeconds()
            var dn="AM"
            if (hours>12){
               dn="PM"
               hours=hours-12
            }
            if (hours==0) hours=12
            if (minutes<=9) minutes="0"+minutes
            if (seconds<=9) seconds="0"+seconds
            document.getElementById("timeValue").innerHTML = hours+":"+minutes+"
"+dn;
            setTimeout("show()",1000)
      }

      function lowAlert() {
          critical = true;
            document.getElementById("ambulanceInfo").style.display = "none";
            document.getElementById("helpInfo").style.display = "block";
            document.getElementById("status").style.background = "#666600";
            document.getElementById("statusValue").innerHTML = lp.ID_37;
            document.getElementById("recommText").innerHTML = lp.ID_38;
            document.getElementById("alertArea").className = "";
            document.getElementById("alertArea").className = "alerts";
```

```
            document.getElementById("alertTxt").className = "";
            document.getElementById("alertTxt").className = "blink";
            document.getElementById("alertTxt").innerHTML = lp.ID_64;
            document.getElementById("alertArea").style.display = "block";
            fetchSendHelpETA(0);
      }

      function hiAlert() {
              critical = true;
              document.getElementById("ambulanceInfo").style.display = "none";
              document.getElementById("helpInfo").style.display = "block";
              document.getElementById("status").style.background = "#666600";
              document.getElementById("statusValue").innerHTML = lp.ID_37;
              document.getElementById("recommText").innerHTML = lp.ID_40;
              document.getElementById("alertArea").className = "";
              document.getElementById("alertArea").className = "alerts";
              document.getElementById("alertTxt").className = "";
              document.getElementById("alertTxt").className = "blink";
              document.getElementById("alertTxt").innerHTML = lp.ID_65;
              document.getElementById("alertArea").style.display = "block";
              fetchSendHelpETA(1);
      }

      function backToNormal() {
            document.getElementById("status").style.background = "#006600";
            document.getElementById("statusValue").innerHTML = lp.ID_42;
            document.getElementById("alertArea").className = "";
            document.getElementById("alertArea").className = "text";
            document.getElementById("alertArea").style.display = "block";
            document.getElementById("alertTxt").className = "";
            refresh = true;
            someEvent = false;
            dprint ("dispatched: "+dispatched);
            if (dispatched != true) {
                  critical = false;
                  document.getElementById("ambulanceInfo").style.display       =
"none";
                  document.getElementById("helpInfo").style.display = "none";
                  document.getElementById("alertTxt").innerHTML = lp.ID_66;
                  $se("stopETAs");
            } else {
                  critical = true;
                  document.getElementById("alertTxt").innerHTML = lp,ID_67;
            }
      }

      function updateStatus() {
         //refresh = true;
         if (latest.condition == 0) {
             lowAlert();
            } else {
                  if (latest.condition == 1) {
                        hiAlert();
                  } else {
                        if (latest.condition == 2) {
                              backToNormal();
                        }
                     }
                }
        }
               function setRefresh() {
            refresh = true;
          }
```

```
        function fetchSendHelpETA(c) {
                //$se("checkForEvent", webView.getWebViewId());
                dprint("Running   fetchSendHelp   on   client.   someEvent   =
"+someEvent);
                if (someEvent == 0) {
                      $se("prepareSendHelp", critical, c);
                }
                $se("subscribeWebviews", webView.getWebViewId());
        }


        function updateSendHelpETA(ETA, d) {
             dispatched = d;
             if (ETA >= 0) {
                 dprint("back to patient_view. ETA,dispatched="+ETA+","+d);
                if                                                    (critical)
document.getElementById("status").style.background = "#666600";
                 document.getElementById("helpInfo").style.display = "block";
               document.getElementById("ambulanceInfo").style.display = "none";
                 document.getElementById("warningETA").innerHTML              =
lp.ID_45+ETA+lp.ID_46;
             }
             if ( (d) && (ETA<=0) ){
                 dispatched = true;
                if                                                    (critical)
document.getElementById("status").style.background = "#660000";
                 document.getElementById("helpInfo").style.display = "none";
               document.getElementById("ambulanceInfo").style.display         =
"block";
                 document.getElementById("ambulanceText").innerHTML = lp.ID_68;
                 document.getElementById("ambulanceValue").innerHTML          =
lp.ID_48+ETA+lp.ID_49;
                 document.getElementById("statusValue").innerHTML = lp.ID_50;
                 document.getElementById("warningETA").innerHTML = "";
            }
        }

        function updateAmbulanceETA(ETA, aa){
             ambulanceArrived = aa;
             if (ETA >= 0) {
                 document.getElementById("ambulanceText").innerHTML = lp.ID_69;
                 document.getElementById("ambulanceValue").style.display      =
"block";
                 document.getElementById("ambulanceValue").innerHTML          =
lp.ID_52+ETA+lp.ID_53;
             }
             if ((aa) && (ETA == 0)) {
                document.getElementById("ambulanceText").innerHTML = lp.ID_54;
                 document.getElementById("ambulanceValue").innerHTML          =
lp.ID_55;
                 ambulanceArrived = true;
                 doorLocked = false;
                 dispatched = false;
                 return;
             }
           return;
        }

        function updateReadings(v,wv) {
         //dprint("back to patient_view. Running updateReadings()...");
         //dprint("wv: "+wv);
         //dprint("this webViewId: "+webView.getWebViewId());
```

```
          if ( webView.getWebViewId() == wv ) {
            latest = JSON.parse(v);

           glucReading  = latest.glucose;
           respReading = latest.respRate;
           exercReading  = latest.exerc;

            //dprint("Do we have any events? "+latest.events);
            someEvent = latest.events;

           document.getElementById("glucValue").innerHTML = glucReading+" mg/dL";
           document.getElementById("respValue").innerHTML = respReading+" bpm";

            if (exercReading < 150) {
            document.getElementById("exercValue").innerHTML = lp.ID_56;
            } else {
                  if ((150 < exercReading) && (exercReading < 300)) {
                  document.getElementById("exercValue").innerHTML = lp.ID_57;
                  } else {
                        if (exercReading > 300) {
                        document.getElementById("exercValue").innerHTML        =
lp.ID_58;
                        }
                     }
               }

           percentGluc = (Math.floor((glucReading*100)/initWidth));
            if (percentGluc > 100 ) percentGluc = 100;
           percentExerc = (Math.floor((exercReading*100)/initWidth));
           percentResp = (Math.floor((respReading*100)/initWidth));
           document.getElementById("gluco").style.width = percentGluc+"%";
           document.getElementById("resp").style.width = percentResp+"%";
           document.getElementById("exerc").style.width = percentExerc+"%";

            if (currCond != latest.condition) {
                  currCond = latest.condition;
                  updateStatus();
            }
          }
        }

         function clearAlerts() {
            $se("clearAlerts");
         }

         function resetAlerts() {
            dispatched = false;
            ambulanceArrived = false;
            doorLocked = false;
            backToNormal();
         }
         function showLive() {
                  document.getElementById("historyData").style.display = "none";
         }

         function showHist() {
            if                (showingLogs                ==                true)
document.getElementById("historyData").style.display = "none";
            if               (showingLogs                ==               false)
document.getElementById("historyData").style.display = "block";

            if (latest.condition == 0) {
```

```
                  document.getElementById("hypoHistory").style.display         =
"block";
                  document.getElementById("hyperHistory").style.display        =
"none";
                  document.getElementById("normalHistory").style.display       =
"none";
            }
            if (latest.condition == 1) {
                  document.getElementById("hypoHistory").style.display = "none";
                  document.getElementById("hyperHistory").style.display        =
"block";
                  document.getElementById("normalHistory").style.display       =
"none";
            }
            if (latest.condition == 2) {
                  document.getElementById("hypoHistory").style.display = "none";
                  document.getElementById("hyperHistory").style.display        =
"none";
                  document.getElementById("normalHistory").style.display       =
"block";
            }
            showingLogs = !showingLogs;
         }

      </SCRIPT>
      <!-- End of <script> Code Block -->
  </HEAD>
  <!-- End of <HEAD> Code Block -->
  <!-- Start of <BODY> Code Block
  The following HTML code displays the web page. The behavior of the
  displayed page is modified by the code in the <script> code block.
  -->
  <BODY onload='init()'>
        <DIV id="app">
           <CENTER>
              <DIV id="ID_txt21" class="text" style="border: 1px solid #bbbbbb;
      height:25px;">Health Care Mobile Monitoring Web Application"</DIV>
           </CENTER>
            <div id="userInfoContainer">
                  <div id="ambIcons">
                        <div     id="clearICN"     onclick="clearAlerts()"><img
src="clear_alerts.png"  width="25px"  height="25px"  alt="Clear  alerts  and  reset
system"></div>
                        <div       id="histICN"       onclick="showHist()"><img
src="history2.png" width="25px" height="25px" alt="5-Hour averages"></div>
                  </div>
                  <div id="userInfo">
                        <div id="userName">Пацієнт: Дід Толя 85 років, діабет 1-
го типу</div>
                  </div>
            </div>
           <!--<CENTER>
            <input type="button" id="confirmAmbulance" value="Confirm Ambulance"
onClick="confirmAmbulance(this.id);"></input>
            <input  type="button"  id="toggleCritical"  value="Toggle  Critical"
onClick="toggleCritical();"></input>
           </CENTER>-->
            <DIV id="status">
                  <div id="pCond">
                        <div><span         id="statusValue"         style="font-
size:17px;">Нормальний</span></div>
                        <div><span  id="ID_txt24"  style="font-size:11px;">Статус
Пацієнта</span></div>
```

```
                </div>
                <div id="time">
                      <div><span          id="timeValue"          style="font-
size:17px;"></span></div>
                      <div><span          id="ID_txt25"          style="font-
size:11px;">Час</span></div>
                </div>
                <div id="helpInfo">
                   <div    id="recommText"    style="font-size:11px;">Insulin
injection advised.</div>
                   <div id="warningETA" style="font-size:11px;">Help to be sent
in: </div>
                </div>
                <div id="ambulanceInfo">
                   <div  id="ambulanceText"   style="font-size:11px;">Ambulance
sent! </div>
                   <div            id="ambulanceValue"            style="font-
size:11px;display:none;">
                      <input class="button" type="button" id="confirmAmbulance"
value="Confirm Ambulance" onClick="confirmAmbulance(this.id);"></input>
                   </div>
                </div>
            </DIV>
            <DIV class="text">
                  <DIV id="gluco" class="meter"></DIV>
                  <DIV   class="labels"><span   id="ID_txt30">Рівень   Глюкози:
</span><span id="glucValue">Безрезультатно</span></DIV>
            </DIV>
            <DIV class="text">
                  <DIV id="resp" class="meter"></DIV>
                  <DIV   class="labels"><span   id="ID_txt32">Частота   Дихання:
</span><span id="respValue">Безрезультатно</span></DIV>
            </DIV>
            <DIV class="text">
                  <DIV id="exerc" class="meter"></div>
                  <DIV class="labels"><span id="ID_txt34">Рівень Навантаження (5-
ти годинне avg.): </span><span id="exercValue">Безрезультатно</span></div>
            </DIV>
            <DIV id="alertArea" class="text">
                  <DIV><span  id="alertTxt"><center>Усі  показники  пацієнта  у
нормі.</center></span></DIV>
            </DIV>
            <div id="footer">
                <DIV id="footerTxt">NTU DP</DIV>
            </div>
        </DIV>
        <div id="historyData" onclick="showLive()">
            <div id="logs" onclick="showLive()">
                  <center><span  id="ID_txt60"  style="font-size:16px;"><p>П'яти-
годинне avg:</p></span><hr></center>
                  <span id="hypoHistory" style="display:none;"><p>Рівень Глюкози:
43mg/dL</p><p>Частота Дихання: 8bpm</p><p>Рівень Навантаження: Високий</p></span>
                  <span    id="hyperHistory"    style="display:none;"><p>Рівень
Глюкози:   298mg/dL</p><p>Частота   Дихання:   25bpm</p><p>Рівень   Навантаження:
Низький</p></span>
                  <span    id="normalHistory"    style="display:none;"><p>Рівень
Глюкози:   97mg/dL</p><p>Частота   Дихання:   15bpm</p><p>Рівень   Навантаження:
Середній</p></span>
            </div>
        </div>
  </BODY>
  <!-- End of <BODY> Code Block -->
</HTML>
```