

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»
Інститут електроенергетики
(інститут)
Факультет інформаційних технологій
(факультет)
Кафедра інформаційних систем та технологій
(повна назва)

ПОЯСНОВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня бакалавра

студента Муштат Олександра Олександровича
(ПІБ)
академічної групи 123-16-1
(шифр)
спеціальності 123 Комп'ютерна інженерія
(код і назва спеціальності)
за освітньо-професійною програмою 123 Комп'ютерна інженерія
(офіційна назва)
на тему Комп'ютерна система фітнес клубу «Спортлайф» з детальним
опрацюванням побудови, налаштування та безпеки корпоративної
мережі.
(назва за наказом ректора)

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	ас. Панферова Я.В.			
розділів:				
апаратний розділ	доц. Ткаченко С.М.			
розрахунок мережі	ас. Панферова Я.В.			
економічний розділ	ст. викл. Яремчук І.О.			
охорона праці	доц. Яворська О.О.			

Рецензент				
-----------	--	--	--	--

Нормоконтролер	проф. Цвіркун Л.І.			
----------------	--------------------	--	--	--

Дніпро
2020

ЗАТВЕРДЖЕНО:
завідувач кафедри
інформаційних систем
та технологій
(повна назва)
Гнатушенко В.В.
(підпис) (прізвище, ініціали)

" _ " _____ 2020 року

ЗАВДАННЯ

на кваліфікаційну роботу ступеня бакалавр
студента Муштат О.О академічної групи 123-16-1
(прізвище та ініціали) (шифр)
спеціальності 123 «Комп'ютерна інженерія»
за освітньо-професійною програмою 123 «Комп'ютерна інженерія»
(офіційна назва)
на тему Комп'ютерна система фітнес клубу «Спортлайф» з детальним
опрацюванням побудови, налаштування та безпеки корпоративної
мережі.

(назва за наказом ректора)

затверджену наказом ректора НТУ «Дніпровська політехніка» від 21.05.2020
№ 771-л

Розділ	Зміст	Термін виконання
Стан питання та постановка завдання	На основі матеріалів виробничих практик, інших науково-технічних джерел сформулювати завдання, конкретизувати предмет та мету роботи	18.05.2020
Технічні вимоги до комп'ютерної системи	На основі матеріалів виробничих практик, інших науково-технічних джерел сформулювати технічні вимоги до розробки комп'ютерної системи	25.05.2020
Спеціальна частина	Розв'язати завдання з розробки комп'ютерної системи з опрацюванням побудови та налаштування корпоративної мережі	01.06.2020
Економічна частина	Економічно обґрунтувати доцільність витрат на створення та дослідження системи	05.06.2020
Охорона праці	Розробити організаційно-технічні заходи щодо реалізації правил безпеки при експлуатації системи	10.06.2020

Завдання видано

_____ (підпис керівника)

ас. Панферова Я.В.

(прізвище, ініціали)

Дата видачі 09.04.2020

Дата подання до екзаменаційної комісії

18.06.2020

Прийнято до виконання _____

Муштат О.О.

РЕФЕРАТ

Пояснювальна записка: ___ с., ___ рис., ___ табл., ___ додатки, ___ джерел.

Об'єкт розробки: корпоративна мережа фітнес клубу «Спортлайф»

Мета кваліфікаційної роботи: модернізація корпоративної мережі фітнес клубу «Спортлайф» з провадженням функцій мережного керування через хмарну платформу Meraki.

Спеціальна частина містить наступні підрозділи: розробка апаратної частини, де представлено обґрунтування вибору топології та технологій, зроблено вибір технічних засобів, розроблена структурна схема та можливості панелі управління Meraki Dashboard.

В розділі «Розробка комп'ютерної ситеми та перевірка її налаштувань» побудована картографічна карта всередині будівлі на платформі Mapwize. Отримані дані від точок доступу Meraki інструментом API. Розроблена схема мережі реалізована у вигляді моделі на базі симулятора Cisco Packet Tracer і перевірена її робота.

У економічній частині приведені розрахунки капітальних витрат на придбання складових КС та на розробку програмного забезпечення.

У розділі «Охорона праці» представлені інженерно-технічні заходи, що стосуються безпеки здоров'я людини.

Розроблене технічне рішення може бути впроваджене на будь-яких об'єктах сервісі послуг, магазинах підприємствах та інших галузях.

MERAKI, MAPWIZE, API, CISCO, МЕРЕЖА, ФІТНЕС-КЛУБ

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	8
1 СТАН ПИТАННЯ І ПОСТАНОВКА ЗАДАЧІ	10
1.1 Характеристика сфери та умов застосування системи, що проектується	10
1.2 Організаційна структура фітнес клубу «Спортлайф»	11
1.3 Структура об'єкта впровадження	13
1.4 Обґрунтування доцільності кваліфікаційної роботи бакалавра	14
1.5 Визначення можливих напрямків рішення поставлених завдань	15
1.6 Використання прикладного програмного інтерфейсу (API) для керування мережевою інфраструктурою	16
1.7 Постанова завдання	18
2 ТЕХНІЧНІ ВИМОГИ ДО КОМП'ЮТЕРНОЇ СИСТЕМИ	19
2.1 Вимоги до системи в цілому	19
2.1.1 Вимоги до структури та функціонування системи	19
2.1.2 Вимоги до складу технічних засобів системи	19
2.1.3 Вимоги до пропускнуої здатності інформаційних каналів	20
2.1.4 Вимоги до чисельності і кваліфікації персоналу, що обслуговує систему і режиму його роботи	20
2.1.5 Вимоги до надійності	21
2.1.6 Вимоги до ергономіки та технічної естетики	21
2.1.7 Вимоги до захисту інформації від несанкціонованого доступу	22
2.1.8 Вимоги до схоронності інформації при аваріях	22
2.1.9 Вимоги до захисту від впливу зовнішніх чинників	22
2.1.10 Вимоги до стандартизації	22
2.2 Вимоги до функцій, які виконує КС	22
2.3 Вимоги до видів забезпечення КС	23

	5
■	23
■	23
2.3.1	Вимоги до інформаційного забезпечення 23
2.3.1.1	Вимоги до інформаційного обміну між компонентами системи 23
2.3.1.2	Вимоги до доступності ресурсів між підсистемами 23
2.3.2	Вимоги до програмного забезпечення 23
3	РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ КОМП'ЮТЕРНОЇ СИСТЕМИ 24
3.1	Обґрунтування вибору топології мережі і технологій 24
3.2	Розробка специфікації апаратних засобів комп'ютерної системи 26
3.2.1	Розміщення точок доступу 28
3.2.2	Відстань між точками доступу 29
3.3	Розробка структурної схеми комплексу технічних засобів системи 30
3.4	Опис і можливості панелі управління Meraki 30
4	РОЗРОБКА КОМП'ЮТЕРНОЇ СИСТЕМИ ТА ПЕРЕВІРКА ЇЇ РОБОТИ 32
4.1	Створення мережі в Meraki Dashboard 32
4.2	Побудова картографічних карт всередині приміщень та імпорт карти в Meraki 33
4.3	Отримання даних від Meraki 36
4.4	Отримання поверхових планів Meraki 36
4.5	Отримання списку точок доступу 37
4.6	Створення конфігуратора 37
4.7	Використання API-інтерфейсу Cisco Meraki Dashboard 38
4.8	Побудова моделі в Packet Tracer та перевірка її роботи 39
5	ЕКОНОМІЧНА ЧАСТИНА 43
5.1	Техніко-економічне обґрунтування розробки 43
5.2	Розрахунок капітальних витрат на придбання складових КС 43
5.3	Розрахунок капітальних витрат на програмне забезпечення 44
5.3.1	Розрахунок часу на розробку програмного забезпечення 44

	6
5.3.2 Розрахунки витрат на розробку програмного продукту	47
Висновок	49
6 ОХОРОНА ПРАЦІ	50
6.1 Загальні положення	50
6.2 Вимоги безпеки перед початком роботи на ПК	54
6.3 Вимоги безпеки під час виконання роботи на ПК	54
6.4 Вимоги безпеки після закінчення роботи на ПК	56
6.5 Вимоги безпеки в аварійній ситуації	56
ВИСНОВКИ	59
ПЕРЕЛІК ПОСИЛАНЬ	60
ДОДАТОК А. JavaScript для імпорту даних з Meraki в Mapwize	62

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

КС	– комп'ютерна система
API	– прикладний програмний інтерфейс
CRM	– платформа управління взаємовідносинами з клієнтами
VLSM	– маски підмережі змінної довжини (англ. Variable Length Subnet Mask);
DHCP	– протокол динамічного налаштування вузла (англ. Dynamic Host Configuration Protocol);
VLAN	– віртуальна локальна мережа (англ. Virtual Local Area Network);

ВСТУП

Швидкий розвиток технологій і цифровізація призводять до того, що побудова мультисервісних мереж з інтеграцією різних послуг є одним з найбільш перспективних напрямків розвитку телекомунікаційних рішень. Будь-яке сучасне підприємство неможливо уявити без телекомунікацій і потужних інформаційних мереж. Крім того, мережі пов'язані з передачею даних і забезпечення безперервності їх роботи, а значить і високої доступності мережі, стає пріоритетною вимогою. В результаті необхідна підтримка нових сервісів, пристроїв, користувачів і т. п., а також і надійний захист від сучасних кіберзагроз. Не є винятком і мережа фітнес-клубів «Спортлайф».

Але подібні вимоги - забезпечення високої доступності та впровадження нових сервісів - суперечать один одному. З одного боку, для виконання першої вимоги мережі потрібна стабільність. З іншого - друга вимога щодо визначення пов'язане зі змінами, тобто з нестабільністю.

Cisco Meraki - це набір мережевих рішень, керованих через Інтернет, який забезпечує єдине джерело управління місцями розташування, інфраструктурою і пристроями. Власне, технології компанії Meraki здатні забезпечувати Wi-Fi, комутацію, безпеку і управління мобільними пристроями за допомогою централізованої хмарної платформи. Meraki привертає підприємства середнього бізнесу тим, що з її допомогою таким підприємствам не доводиться розширювати штат ІТ-відділу, щоб будувати власні мережі.

Актуальність даної теми зумовлена впровадженням більш гнучкої системи управління мережевою інфраструктурою, розуміння в рамках компанії: хто, куди, коли, заходив і скільки даних передав.

Завдяки платформі Meraki розширюються можливості ІТ-відділів, які зможуть швидше і ефективніше реагувати на зміни і вирішувати виникаючі завдання.

Метою роботи є надання високоякісних послуг клієнтам фітнес клубу «Спортлайф» за рахунок побудови сучасної телекомунікаційної мережі.

1 СТАН ПИТАННЯ І ПОСТАНОВКА ЗАДАЧІ

1.1 Характеристика сфери та умов застосування системи, що проектується

Фітнес клуби – це споруди, які мають площі для проведення оздоровчих та фітнес-тренувань за допомогою силових вправ і обладнані для кардіо-тренувань, які відкриті для вільного відвідування за плату на основі платежів за разове відвідування або по членській системі.

«Спортлайф» – це сучасна мережа фітнес-клубів. Основним напрямом діяльності компанії є розширення мережі фітнес-центрів та спортивних клубів, що працюють відповідно до світових стандартів. Сьогодні група компаній Sport Life управляє мережею фітнес-клубів, магазинів спортивного одягу, інвентаря і соляріїв. По Україні відкрито більше 65 клубів і продовжується розвиватися мережа фітнес-клубів. Для того щоб забезпечити комфортне проведення часу кожного клієнта, в кожному клубі побудована локальна обчислювальна мережа, яка забезпечує відвідувачів безкоштовним гостьовим Wi-Fi.

Фітнес клуб «Спортлайф» працює на основі членської системи річних абонементів. «Спортлайф» – широко профільний фітнес клуб, який включає в себе: тренажерну залу, зали групових програм, аква зону, лаунж зону та дитячий клуб. Тренажерний зал включає в себе кардіозону, зону вільних ваг, зону хамерів, важільних та блокових тренажерів, зону стрейчінгу. В залах для групових програм проводяться групові заняття з пілатесу, йоги, степ аеробіки, велогонки, боксу, танців та інших програм. Аква зона включає в себе басейн для дорослих, дітей та немовлят. Також в аква зоні проводяться різноманітні групові заняття для всіх вікових груп клієнтів.

Для дітей віком до 14 років працює дитячий клуб у якому вихователі проводять з дітьми спортивно-оздоровчі та інтелектуально-розвиваючі заняття та майстер класи.

Для відпочинку та відновлення сил, для клієнтів доступні римська парна, фінська сауна та дві російські лазні.

Для клієнтів надається можливість додатково займатись з персональними тренерами з боксу, аеробіки, йоги, тренажерного залу та плавання.

1.2 Організаційна структура фітнес клубу «Спортлайф»

Фітнес клуб включає в себе ряд технічних та адміністративних відділів. Окремо слід виділити тренажерний зал, басейн, дитячий клуб та групові програми. В цих відділах працюють інструктори, які проводять персональні тренування та групові заняття і підпорядковуються менеджерам даних підрозділів.

До адміністративних відділів відноситься: відділ сервісу, відділ продажів, відділ служби безпеки, центральна рецепція клубу (адміністрація), дирекція та бухгалтерія. До технічних відноситься: відділ технічної експлуатації та відділ клінінгу.

Відділ сервісу працює з діючими членами клубу, основна задача відділу – вирішення конфліктних ситуацій, робота зі скаргами і побажанням клієнтів, продовження абонементів діючих клієнтів, інформування клієнтів про актуальні акції. У відділі працюють сервіс-менеджери, які підпорядковані старшому сервіс-менеджеру.

Відділ продажів працює з потенційними клієнтами, його основна задача – продаж абонементів новим клієнтам. У відділі працюють сейл-менеджери (менеджери з продажів), які підпорядковані старшому сейл-менеджеру.

Служба безпеки клубу контролює роботу всіх підрозділів клубу, забезпечує порядок і безпеку клієнтів в клубі, фіксує факти порушень клубних правил. Співробітник служби безпеки підпорядковується старшому співробітнику зміни, а старший співробітник в свою чергу – начальнику служби безпеки.

Центральна рецепція клубу займається реєстрацією візиту клієнта до клубу, оформленням платних та безкоштовних послуг, забезпечує взаємодію всіх підрозділ між собою. Адміністратор центральної рецепції підпорядковується старшому адміністратору.

До дирекції відноситься фітнес-директор та його асистент, яким підпорядковані усі структурні підрозділи клубу, крім служби безпеки. Дирекція контролює роботу всіх структурних підрозділів, відповідає за матеріальне забезпечення клубу, виконання планів з надання послуг.

У відділі технічної експлуатації працюють інженери з експлуатації, основна задача, яких – підтримання належного функціонування усіх систем та обладнання клубу. Інженери з експлуатації підпорядковані управляючому.

Співробітники відділу клінінгу забезпечують підтримання чистоти та забезпечення дотримання санітарно-гігієнічних норм на території клубу. Співробітники клінінгу підпорядковані клінінг-менджеру.

Організаційна структура фітнес клубу «Спортлайф» наведена на рисунку 1.1.

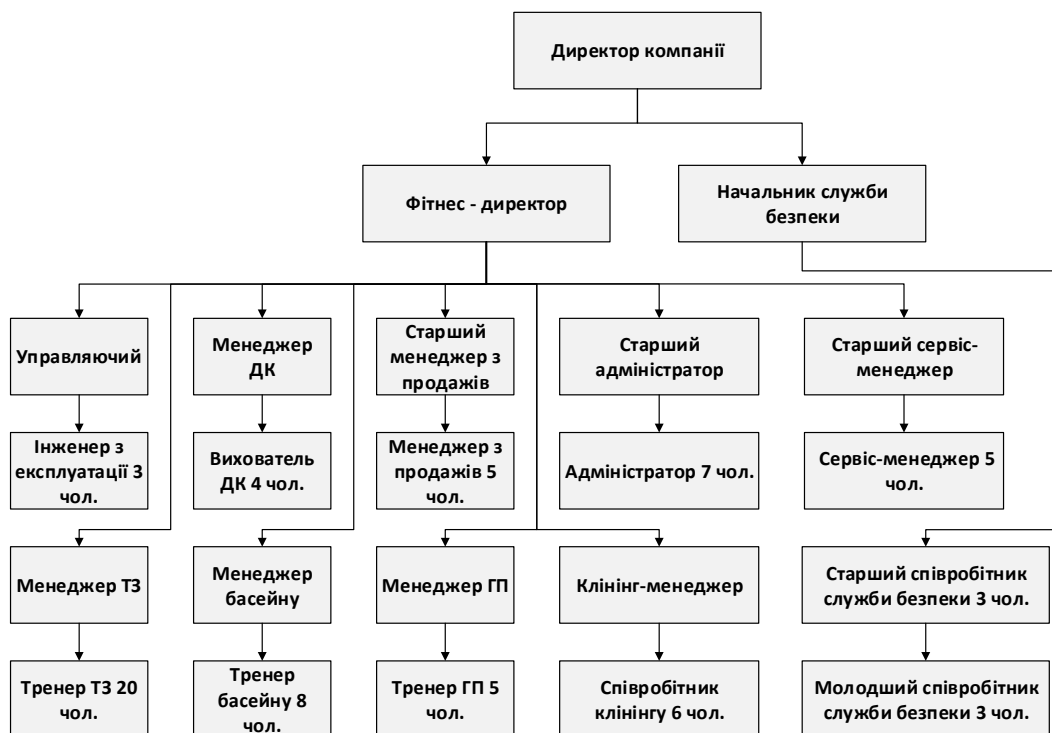


Рисунок 1.1 – Схема організаційної структури фітнес клубу «Спортлайф»

1.3 Структура об'єкта впровадження

Для забезпечення нормального функціонування фітнес клубу технологічний комплекс включає в себе: систему кондиціонування повітря, центральну аудіосистему для оповіщення клієнтів, систему відеонагляду, систему протипожежної безпеки, система внутрішнього телефонного та стільникового зв'язку, систему водо підготовки води для аква зони, систему обігріву бань, сауни та парної, комп'ютерну систему для роботи з клієнтами.

План будівлі фітнес клубу «Спортлайф Приозерний» в м. Дніпро за адресою вул. Боброва 1, наведений на рисунку 1.2.

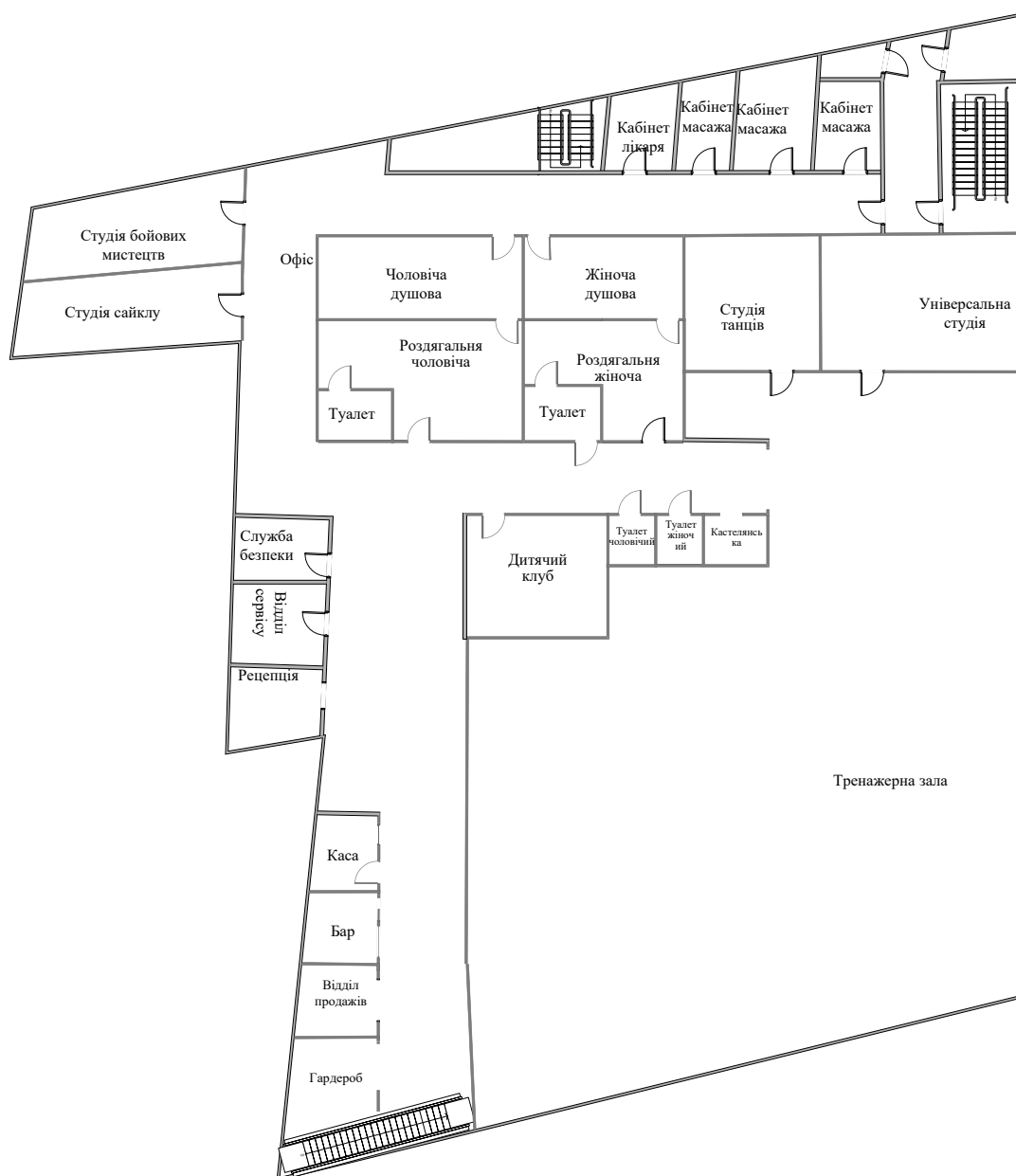


Рисунок 1.2 – План будівлі фітнес клубу «Спортлайф Приозерний»

Комп'ютерна система для роботи з клієнтами поєднує між собою центральну рецепцію, відділ сервісу та відділ продажів. Технологічний комплекс даних підрозділів складається з центрального серверу, на якому наявна клієнтська база клубу, автоматизованих робочих місць менеджерів та адміністраторів, а також мережевого обладнання стандарту Ethernet. Робочі місця співробітників оснащені персональними комп'ютерами та зчитувачами клієнтських карток, які підключені до локальної комп'ютерної мережі фітнес клубу і мають доступ до мережі Інтернет. Менеджери з продажів додають нових клієнтів до бази даних, а сервіс-менеджери та адміністратори використовують існуючу базу для роботи.

1.4 Обґрунтування доцільності кваліфікаційної роботи бакалавра

Компанія планує модернізацію, розширення сфери надаваних послуг і сервісів, модернізацію бізнес-процесів. Необхідно організувати коректний облік робочого часу співробітників, а також спростити процес верифікації телефонних номерів відвідувачів та підвищити його ефективність. Крім того, компанія має намір реалізувати проактивну політику роботи з клієнтами.

Для вирішення цієї проблеми потрібно проаналізувати та впровадити сучасні мережні рішення, які б дозволяли спостерігати за подіями в системі в режимі реального часу, а також отримувати всілякі звіти з моніторингу та геопозиціонування, тепловими картками клієнтів, відвідуваності різних зон, і т.д.

Також в зв'язку з розширенням мережі клубів доводиться розширювати штат ІТ-відділу, щоб будувати власні мережі. Велика частина налаштувань, пов'язаних зміною таких служб, як DHCP, правила переадресації портів, пул підмереж і параметри лінії зв'язку, тягне за собою неоднократне повторювання цих налаштувань на кожному пристрої, що приводить до помилок внаслідок людського. Мережеві адміністратори витрачають багато часу на ці повторювані завдання налаштування і

моніторингу, які можна автоматизувати і зробити без помилок за допомогою API. Оскільки все більше доводиться керувати великою кількістю мереж, є потреба в нових способах підвищення ефективності налаштування та керування існуючими та новими мережами філіалів.

Для вирішення цих проблем потрібно проаналізувати та впровадити сучасні надійні та безпечні мережні рішення.

1.5 Визначення можливих напрямків рішення поставлених завдань

Одним з кращих рішень для внутрішнього позиціонування є використання Wi-Fi. Смартфони з Wi-Fi тепер можна використовувати як індикатор присутності клієнта завдяки механізму Wi-Fi, який є загальним для всіх таких пристроїв: перевірочні запити. Ці кадри управління 802.11 передаються через рівні проміжки часу від пристроїв Wi-Fi. Кадри містять інформацію, яку можна використовувати для визначення присутності, витраченого часу і повторних відвідувань в межах зони доступу Wi-Fi. Ці пристрої можуть бути виявлені точками доступу Wi-Fi незалежно від стану зв'язку Wi-Fi, що означає, що навіть якщо користувач не підключає свій пристрій до бездротової мережі, його присутність все ще можна виявити, коли пристрій знаходиться в межах досяжності мережі і антена Wi-Fi пристрою включена.

Завдяки швидкому впровадженню мобільних пристроїв багато організацій тепер можуть використовувати ці дані для кращого розуміння моделей і поведінки клієнтів в звичайному середовищі. Ця інформація про місцезнаходження, заснована переважно на стандартах бездротового зв'язку і bluetooth 802.11, може використовуватися для залучення користувачів і оптимізації маркетингових стратегій та бізнес-процесів.

Компанія Cisco є лідер в провадженні нових інновацій та технологій. У листопаді 2012 року Cisco Systems за 1,2 мільярда доларів придбала компанію Meraki, яка займається рішеннями в сфері віддаленого управління

комп'ютерними мережами різних масштабів.

Служба визначення місця розташування безпосередньо доступна для точок доступу Cisco Meraki і може бути активована для всіх інших точок доступу Cisco Unified Wireless Networking з використанням рішення Connected Mobile Experiences (CMX). Завдяки API-інтерфейсів інфраструктури, які Cisco робить доступними для розробників, дані про місцезнаходження доступні і можуть використовуватися для позиціонування на карті будь-якого об'єкта з підтримкою Wi-Fi. Точки бездротового доступу Meraki і хмарна інфраструктура збирають ці дані і представляють їх в сукупності на панелі інструментів Meraki. Це робиться за допомогою інтуїтивно зрозумілих і настроюваних графіків, які можна використовувати для розуміння таких тенденцій, як коефіцієнт заповнення (кількість відвідувачів), залученість користувачів (загальний проведений час) і лояльність відвідувачів (нові та повторні відвідування). Meraki може надати цю аналітику всім організаціям, використовуючи передову хмарну архітектуру, яка стоїть за всіма продуктами Cisco Meraki. Крім того, Meraki Scanning API здатний експортувати необроблені дані з спостережуваних запитів зондів, які організації можуть використовувати для безпосередньої інтеграції зі сторонніми платформами сховищ даних або аналітики. Це не тільки може сприяти більш глибокій інтеграції з традиційними платформами управління взаємовідносинами з клієнтами (CRM), але завдяки своїй природі в реальному часі відкриває нові можливості для залучення нових клієнтів.

1.6 Використання прикладного програмного інтерфейсу (API) для керування мережевою інфраструктурою

У сучасному світі дуже розвинені сфери послуги подання, отримання і передачі інформації. Для взаємодії та впровадження нових сервісів широко використовується метод через API.

Прикладний програмний інтерфейс (інтерфейс програмування застосунків, інтерфейс прикладного програмування) (англ. Application Programming Interface, API) – набір визначень підпрограм, протоколів взаємодії та засобів для створення програмного забезпечення [5]. API надає розробнику засоби для швидкої розробки програмного забезпечення. API може бути для веб-базованих систем, операційних систем, баз даних, апаратного забезпечення, програмних бібліотек. Використовується програмістами при написанні всіляких додатків.

Cisco Meraki Dashboard API є RESTful API, який використовує HTTPS запити і JSON в якості формату. Використання структурованих мережевих даних полегшує інтеграцію з системами бізнес-логіки. При створенні API вони можуть вибирати з більш широкого кола кваліфікованих ресурсів. Підприємства можуть створити свою панель моніторингу для менеджерів магазинів або конкретних варіантів використання, що дозволяє їм швидко додавати нові послуги.



Рисунок 1.3 – Передача даних між різними пристроями

Meraki має ієрархічну структуру. У верхньому рядку знаходиться об'єкт «Організація». Це логічний контейнер для мереж Meraki, керованих однією або декількома обліковими записами. Під ним може бути безліч мереж. У середині цих мереж знаходяться мережеві пристрої (пристрої безпеки, комутатори, точки доступу та інтелектуальні камери), які містять атрибути та інформацію, які є керовані і налаштовуються через веб-браузер

Meraki Dashboard. Вся ця інформація, а також параметри конфігурації пристрою зберігаються в хмарі Meraki

1.7 Постановка завдання

Завдання даної кваліфікаційної роботи – модернізація корпоративної мережі фітнес клубу «Спортлайф» з провадженням функцій мережного керування через хмарну платформу Meraki.

Мета даної кваліфікаційної роботи – підвищення ефективності налаштування та керування існуючими та новими мережами філіалів, збір інформації в режимі реального часу з моніторингу та геопозиціонування, тепловими картками клієнтів, відвідуваності різних зон, і т.д.

Впровадження проекту дозволить оптимізувати діяльність закладів, забезпечити оперативний доступ до даних, модернізувати бізнес-процеси та реалізувати проактивну політику роботи з клієнтами.

Для досягнення поставленої мети дипломної магістерської роботи сформульовані наступні завдання:

- обґрунтувати вибір топології мережі і технологій;
- розробити специфікацію апаратних засобів КС;
- розробити структурну схему комплексу технічних засобів КС;
- дослідити можливості платформи Meraki;
- розробити картографічну карту всередині приміщень;
- побудувати модель в Packet Tracer та перевірити її роботу.

2 ТЕХНІЧНІ ВИМОГИ ДО КОМП'ЮТЕРНОЇ СИСТЕМИ

2.1 Вимоги до системи в цілому

2.1.1 Вимоги до структури та функціонування системи

Сервіси «Спортлайф» потребують впровадження наступних функцій:

- віртуалізація мережі за допомогою апаратного і програмного забезпечення на платформі Meraki;
- автоматизація мережі для узгодження застосування політик;
- автоматичне управління мережею в режимі реального часу, контроль стану і працездатності мережі та оповіщення про події та помилки;
- швидке розгортання з оптимізацією апаратних ресурсів;
- динамічний розподіл радіоканалів і оптимізація потужності точок доступу;
- можливість виявлення фізичного місця розташування користувачів, підключених до системи бездротового доступу;
- інтеграцію з соціальними мережами.

2.1.2 Вимоги до складу технічних засобів системи

Бездротова мережа побудована на продуктах Cisco з управлінням з хмарної платформи Meraki .

Вимоги до хмарно-керованих точок доступу:

- самоконфігурація, розгортання за принципом «включай і працюй»;
- протоколи 802.11ac і 802.11n MIMO з підтримкою до трьох просторових потоків для голосу і відео;
- інтегрована безпека підприємства і гостьовий доступ;
- виділене радіо для безпеки і оптимізації RF з інтегрованим аналізом спектра;
- інтегрована система виявлення і запобігання вторгнень (WIDS/WIPS);
- самонавчальний механізм аналізу трафіку з урахуванням додатків;

- гнучкий механізм групової політики для створення і застосування політик з урахуванням додатків по мережі, типу пристрою і кінцевого користувачева;
- адміністрування на основі ролей і автоматичні, заплановані оновлення прошивки, що надаються через Інтернет;
- попередженнями на електронну пошту чи текстові повідомлення про збої живлення, простоях або зміни конфігурації;
- привабливий дизайн і компактні розміри.

Технічне забезпечення системи повинно максимально і найбільш ефективним чином використовувати існуючі технічні засоби.

Активне мережеве обладнання комп'ютерної мережі забезпечує:

- високошвидкісний обмін даними між розеткою підключення обладнання і/або серверами за технологією Ethernet (Fast Ethernet, Gigabit Ethernet) та Wi-Fi 802.11n частотою 2,4 та 5 ГГц;
- ефективну роботу всіх складових елементів локальної обчислювальної мережі ЛОМ (АРМ, серверів, принтерів і т.п.).

2.1.3 Вимоги до пропускної здатності інформаційних каналів

Повинно використовувати віту пару категорії 5е, який нараховує чотири пари, використовується при конструюванні мережі 100/1000 Мбіт/с. Коли взаємодіють дві пар, швидкість передачі - 100Мбіт/с, якщо використовують всі чотири пари – 1000Мбіт/с. Частотна полоса 100 МГц.

Бездротова мережа забезпечувати пропускну здатність не менше 600 Мбіт/с.

2.1.4 Вимоги до чисельності і кваліфікації персоналу, що обслуговує систему і режиму його роботи

Чисельність обслуговуючого персоналу: 1 системний адміністратор та з кваліфікацією в сфері інформаційних технологій.

Графік стандартний 8 годинний робочий день 5днів в тиждень.

2.1.5 Вимоги до надійності

Безперебійна робота серверного обладнання (24x7x365). У разі повного відключення електроенергії система повинна функціонувати протягом 60 хв.

Безперебійна робота локальної мережі, комутаційних шаф, де знаходиться активне мережеве обладнання. У разі повного відключення електроенергії мережа повинна функціонувати протягом 30 хв.

1. У ЛОМ повинні бути застосовані технічні засоби, що мають середній час безвідмовної роботи обладнання ЛВС не менше $T_0 = 35000$ годин. При цьому середній час відновлення працездатності ЛОМ має бути не більше $t_v = 0,4$ години.

2. Всі бази даних повинні бути розміщені на основному апаратній сервері, який для підвищення готовності та відмовостійкості повинен мати додатковий диск для резервного копіювання.

2.1.6 Вимоги до ергономіки та технічної естетики

Елементи комп'ютерної мережі мають бути промарковані. Маркування має бути нанесено :

- на обох кінцях кабелю;
- на телекомунікаційних розетках;
- на комутаційних панелях;

Кожне автоматизоване робоче місце повинно складатися з інформаційної розетки RJ-45 в кількості 2 штуки.

Нумерація розеток наноситься на креслення розташування робочих місць. Кожен порт RJ-45 повинен бути промаркований номером кімнати і через «.» номером порту у кімнаті.

Зовнішній вигляд коробів та аксесуарів повинен гармоніювати з інтер'єром робочих місць.

2.1.7 Вимоги до захисту інформації від несанкціонованого доступу

Забезпечити можливість безпечного та захищеного віддаленого адміністрування через мережу Інтернет.

Забезпечити обмеження доступу до серверного обладнання з корпоративної мережі та мереж загального користування.

Забезпечити мережевий захист від вірусних атак, DDoS.

2.1.8 Вимоги до схоронності інформації при аваріях

Забезпечити резервне копіювання даних які підлягають довготривалому зберіганню (передбачити схему резервного копіювання в проекті).

2.1.9 Вимоги до захисту від впливу зовнішніх чинників

Всі кабельні з'єднання повинні знаходитися в кабель каналах для захисту від зовнішнього впливу.

2.1.10 Вимоги до стандартизації

Усе активне обладнання повинно мати сертифікат відповідності. Для створення локальної обчислювальної мережі необхідно використовувати тільки високоякісні компоненти, які пройшли стовідсоткове тестування відповідно до вимог ISO 9001 (ГОСТ 40.9001-88).

2.2 Вимоги до функцій, які виконує КС

Побудова єдиного інформаційного середовища фітнес-центру для забезпечення наступних функцій:

- управління мережним обладнанням з хмарної платформи Meraki;
- збір інформації про місцезнаходження відвідувачів в режимі реального часу;

– отримувати всілякі звіти з моніторингу та геопозиціонування, тепловими картками клієнтів, відвідуваності різних зон, і т.д.

2.3 Вимоги до видів забезпечення КС

2.3.1 Вимоги до інформаційного забезпечення

2.3.1.1 Вимоги до інформаційного обміну між компонентами системи

Інформаційний обмін між компонентами КС повинен здійснюватися за допомогою захищених протоколів через стек протоколів TCP/IP.

2.3.1.2 Вимоги до доступності ресурсів між підсистемами

Доступність інформаційних ресурсів між компонентами підсистем повинна забезпечуватися за рахунок надійного доступу до Інтернет.

2.3.2 Вимоги до програмного забезпечення

Програмне супроводження та обслуговування мережевого варіанту програмного комплексу через хмару в веб-браузері або API-інтерфейси програмування.

3 РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ КОМП'ЮТЕРНОЇ СИСТЕМИ

3.1 Обґрунтування вибору топології мережі і технологій

Згідно з вимог розроблене рішення повинно бути впроваджено на платформі Meraki. Meraki – це 100% хмарне рішення для побудови IT-інфраструктури невеликої або середньої компанії, яким характерно швидке розгортання, спрощене адміністрування і багатофункціональність.

Сервіс Cisco Meraki включає в себе бездротову інфраструктуру, дротову інфраструктуру, Meraki MC для для голосового зв'язку елементи забезпечення безпеки з можливістю централізованого управління та відеокамери за допомогою хмарних технологій (рис. 3.1).

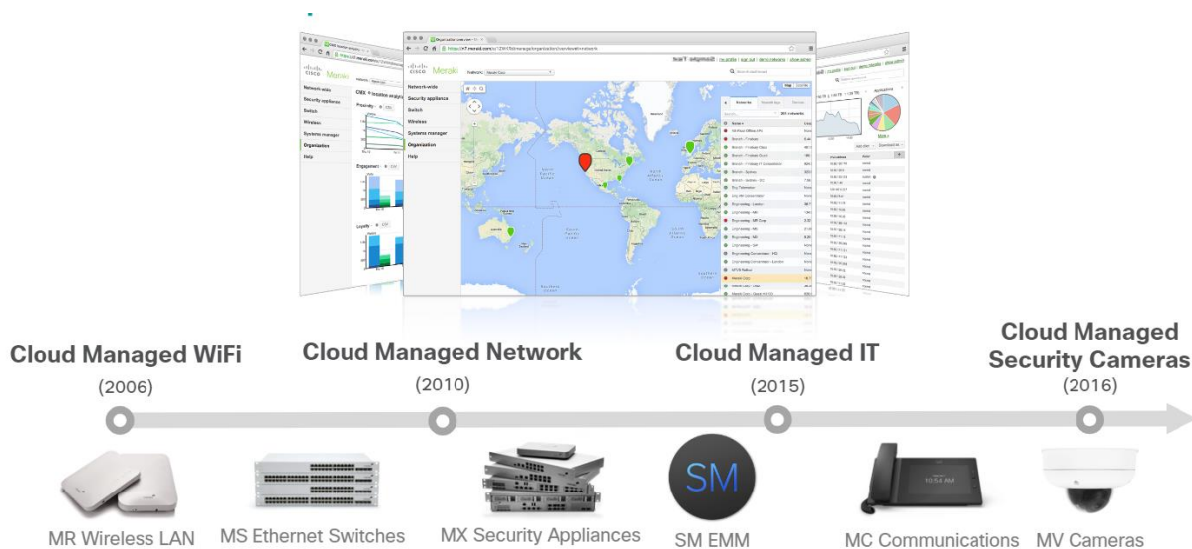


Рисунок 3.1 – Інфраструктура Meraki

Для організації повноцінного Wi-Fi рішення слід встановити точки доступу Meraki і під'єднати до центральної cloud-системі управління з використанням надійних Інтернет каналів зв'язку. Cloud-рішення не вимагає наявності апаратного контролера точок доступу і зменшує початкові витрати на розгортання системи.

Панель управління Meraki (Meraki Dashboard) є частиною архітектури хмарного управління Cisco Meraki, яка дозволяє візуалізувати мережеву інфраструктуру з демонстрацією даних про всі підключені пристрої,

статистики мережевого трафіку і моніторингу стану мережі. За допомогою панелі керування адміністратори можуть додавати нові пристрої і автоматизувати такі рутинні завдання, як знаходження і установку останніх версій прошивок. Всі пристрої Cisco Meraki підключаються до хмари в автоматичному режимі, використовуючи протокол SSL.

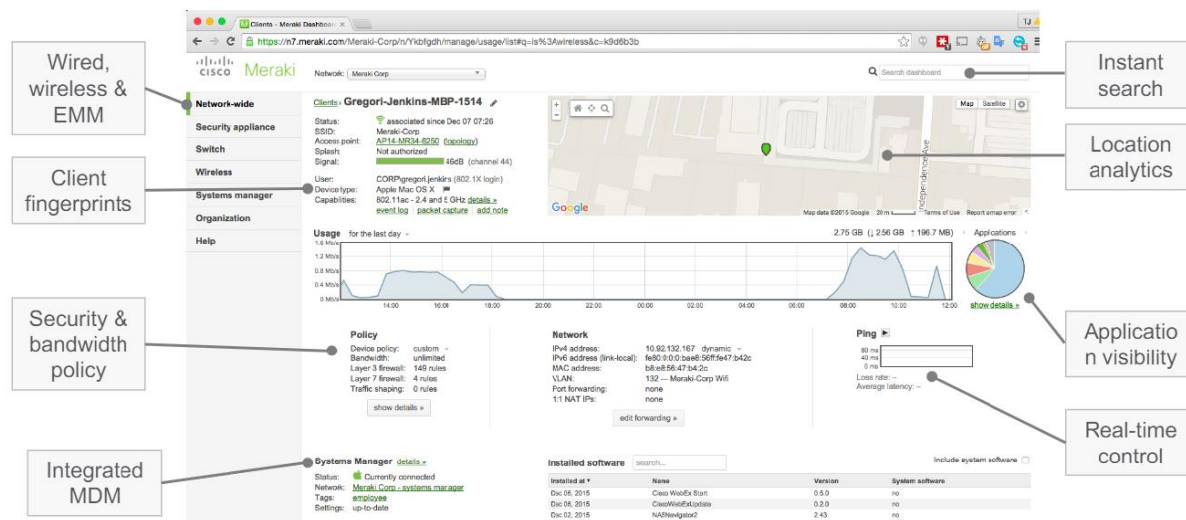


Рисунок 3.2 – Керування через Meraki Dashboard

Користувачі можуть розгортати, контролювати і налаштовувати свої пристрої Meraki через веб-інтерфейс панелі інструментів Meraki або через API. Як тільки користувач вносить зміну в конфігурацію, запит на зміну відправляється в хмару Meraki, а потім вирушає з хмари Meraki на відповідний пристрій (пристрої).

Керуючі дані (наприклад, конфігурація, статистика, моніторинг і т. д.) передаються з пристроїв Meraki (точок бездротового доступу, комутатори та пристроїв безпеки) в хмару Meraki по захищеному інтернет-з'єднанню.

Данні користувача (перегляд веб-сторінок, внутрішні програми і т. д.) не передаються через хмару Meraki, а прямують безпосередньо до місця призначення в локальній мережі або через глобальну мережу (рис. 3.3).

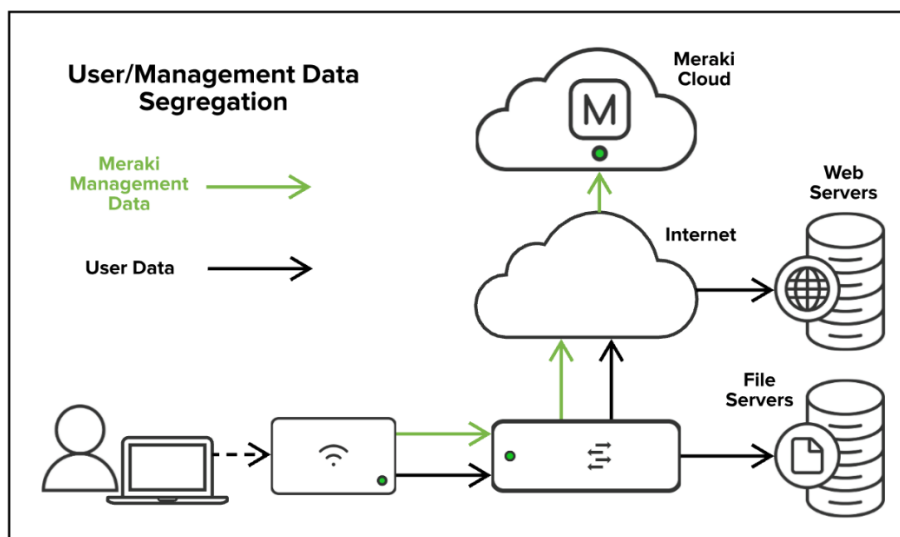


Рисунок 3.3 – Архітектура Meraki

Пристрої для зв'язку з хмарою Meraki використовують запатентований легкий зашифрований тунель, який використовує шифрування AES256. У самому тунелі Meraki використовує HTTPS і протокольні буфери для безпечного і ефективного вирішення, обмеженого 1 кбіт/с на пристрій, коли пристрій активно не керується.

3.2 Розробка специфікації апаратних засобів комп'ютерної системи

Дослідивши бездротові точки доступу, які застосовані для створення мережі Wi-Fi в закладі, було виявлено головний їх недолік: вони не підтримують віртуалізацію мережі. Тому необхідно обрати точки доступу Wi-Fi з можливістю підключення до Meraki.

Рішення Cisco Meraki для бездротових локальних мереж включають такі точки доступу Cisco Meraki:

- внутрішні: моделі MR18, MR32, MR34, MR42, MR52, MR53;
- зовнішні: моделі MR62, MR66, MR72, MR74, MR84.

Обираємо MR34 для внутрішнього застосування. Модель Meraki MR34 виробництва Cisco представляє собою новітню точку доступу стандарту 802.11ac з пропускнуою спроможністю, яка в три рази вище швидкості стандарту 802.11n.



Рисунок 3.4 – Wi-Fi точка доступа Cisco Meraki MR34

Точка доступу MR34 здатна забезпечити передачу даних на швидкості в 1,75 Gbps. Більш того, пристрій був розроблений для чутливих в сенсі безпеки мереж, а тому має не два а три радіомодуля, один з яких виділено спеціально для сканування двох підтримуваних частотних діапазонів на предмет можливих ризиків проникнення або блокування сигналу. Хмарні ж сервіси MR34 дозволяють не тільки дистанційно керувати точкою доступу, але також сайтами, рекламою, аналізувати дані про підключених користувачів. Характеристики надані в таблиці 3.1.

Таблиця 1.1 – Характеристики MR34

Апаратні особливості	3 радіо: 2,4 і 5 ГГц, дводіапазонні WIDS / WIPS 3-потік 802.11ac і 802.11n, до 1,75 Гбіт 1 × 100 / 1000Base-T Ethernet (RJ45) 48V DC 802.3at / 802.3af PoE 1 × DC power connector (5mm x 2.1mm, center positive)
Хмарне управління	Прозорість і керованість всієї мережі Самоадміністрування для швидкого розгортання автоматична звітність Безшовні оновлення прошивки
Корпоративна безпека	802.1X і вбудовану інтеграцію Active Directory Integrated policy firewall (Identity Policy Manager) Air Marshal: real-time WIPS with forensics Guest Isolation WEP, WPA WPA2-PSK WPA2-Enterprise with 802.1X TKIP and AES encryption VLAN tagging (802.1Q) Stateful Layer 3-7 брандмауера Групові політики на основі ідентифікаційної інформації Вбудована антивірусна перевірка (NAC)
Гостьовий доступ	Безпечний гостьовий доступ Настроюються сторінки сплеску інтегрований Facebook

В якості модернізації і розширення можливостей мережі рекомендується оновлення комутаторів доступу на модель MS320, а комутатори рівня розподілу на MS420.

3.2.1 Розміщення точок доступу

Правильне розміщення точок доступу рекомендується для того, щоб повною мірою використовувати переваги точності визначення місця розташування. Важливо відзначити, що всі точки доступу не повинні бути згруповані усередині плану поверху. Замість цього AP можуть бути розміщені по периметру плану поверху, щоб забезпечити постійне покриття на всьому протязі. Додаткові точки доступу можуть бути розміщені в кутах плану поверху, щоб підвищити точність визначення місця розташування для клієнтських пристроїв. Ці кутові точки доступу грають життєво важливу роль в забезпеченні високої точності визначення місця розташування клієнтів, які перебувають всередині периметра. Приклади такого розміщення показані на зображеннях в плані нижче. Розміщення точок доступу (рис. 3.5).

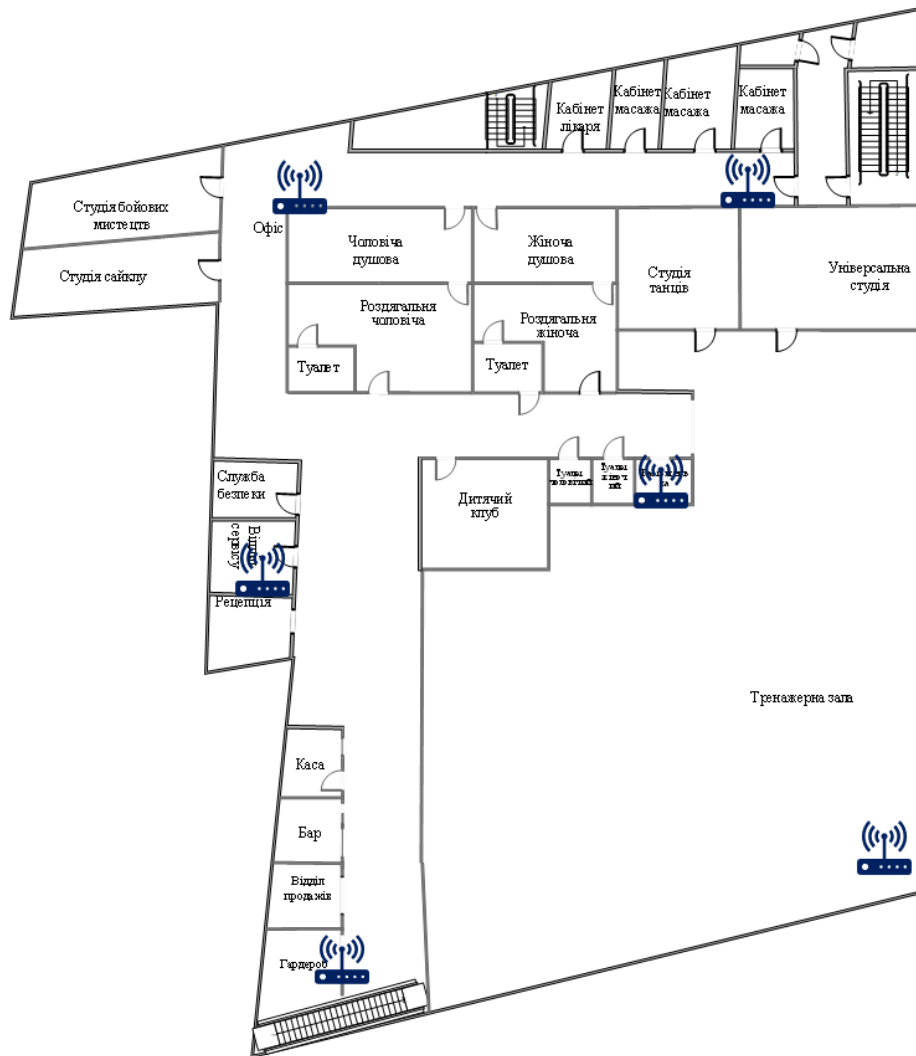


Рисунок 3.5 – Розміщення точок доступу

3.2.2 Відстань між точками доступу

Розташування точок доступу може вплинути на відстеження місця розташування клієнта, а також на продуктивність бездротового зв'язку в цілому. Тому рекомендується, щоб точки доступу не розташовувалися занадто близько один до одного або занадто далеко один від одного. Гарантуйте, що значення SNR для клієнта не опускається нижче 20 дБ, оскільки вони переміщуються між AP. Крім того, мінімум три точки доступу (з чотирма або більше для кращої точності і точності) повинні мати можливість декодувати клієнтські кадри в будь-який момент часу для відстежування місцеположення клієнта.

3.3 Розробка структурної схеми комплексу технічних засобів системи

Структурна схема комплексу технічних засобів комп'ютерної системи зображена на рисунку 3.4. Схема виконана в узгодженні структури з топологічними особливостями об'єкту розробки та є найбільш доцільною та продуктивною, а також виконує всі поставлені перед системою вимоги та задовольняє кількісний склад технічних засобів.

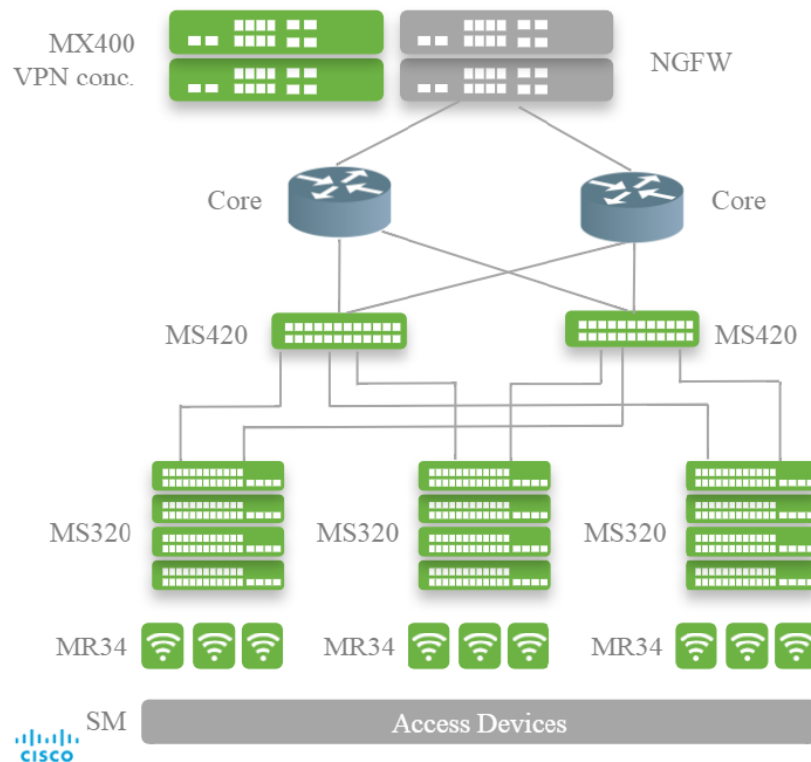


Рисунок 3.6 – Структурна схема комплексу технічних засобів системи

Компанія Cisco представляє архітектуру Cisco Meraki з панеллю керування Meraki Dashboard, яка дозволяє значно підвищити продуктивність адміністрування і роботи мереж за допомогою використання хмари.

3.4 Опис і можливості панелі управління Meraki

При використанні архітектури Meraki обладнання розміщується в головному офісі, після чого пристрою Cisco Meraki (точки доступу,

комутатори та пристрої безпеки) підключаються до хмари з використанням шифрування SSL.

Панель управління Dashboard дозволяє візуалізувати всі мережеві ресурси Meraki і спрощує процес мережевого менеджменту. Зокрема, можна швидко додавати нові пристрої, завантажувати конфігурації і прошивки в автоматичному режимі, а також виконувати моніторинг стану мережі.

Дане рішення є єдиною точкою управління, консолідує всі засоби автоматизації і моніторингу. Такий підхід значно спрощує роботу адміністраторів.

Безпека гарантується використанням шифрованого каналу для зв'язку з хмарою. При цьому для підключення використовується подвійна аутентифікація.

Хмара є з декількох ЦОД. Навіть при втраті зв'язку з хмарою, робота мережі здійснюється в безперервному режимі.

Використання архітектури Meraki виключає необхідність використання апаратних контролерів і комплексів управління.

4 РОЗРОБКА КОМП'ЮТЕРНОЇ СИСТЕМИ ТА ПЕРЕВІРКА ЇЇ РОБОТИ

4.1 Створення мережі в Meraki Dashboard

Як було сказано, панель управління Meraki Dashboard дозволяє візуалізувати всі мережеві ресурси Meraki і спрощує процес мережевого менеджменту. Зокрема, можна швидко додавати нові пристрої, завантажувати конфігурації і прошивки в автоматичному режимі, а також виконувати моніторинг стану мережі.

Щоб почати, спочатку потрібно створити обліковий запис на Meraki Dashboard. Для цього потрібно перейти на сайт <https://dashboard.meraki.com> та створити обліковий запис.

Потім, коли є обліковий запис і організація Meraki, можна створити мережу. Мережі використовуються для зберігання пристроїв і їх конфігурацій. Звичайною практикою є створення мережі пристроїв для кожного фізичного місця розташування.

Щоб створити мережу обладнання або пристрої, в меню обрати Organization->Create Network (рис. 3.7).

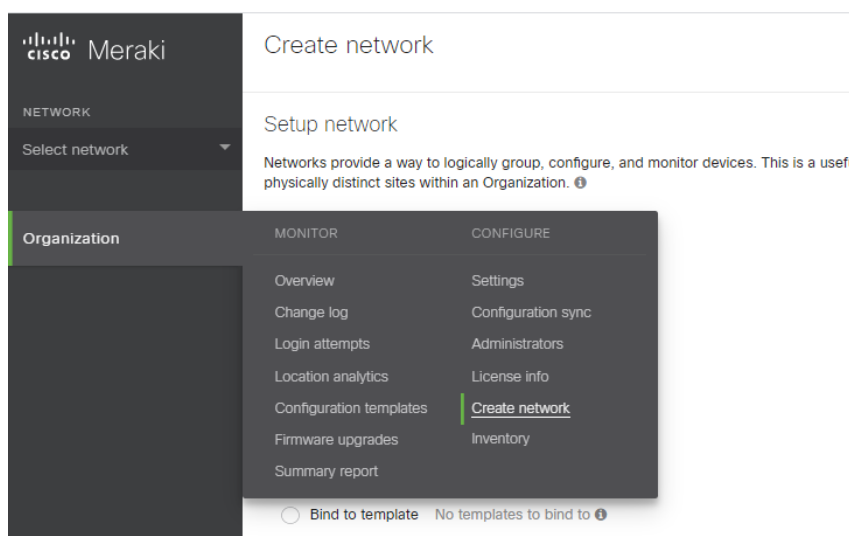


Рисунок 4.1 – Меню створення мережі

Рисунок 4.2 – Створення нової мережі в Meraki Dashboard

Network Name – ім'я, яке буде використовуватися для ідентифікації цієї мережі на панелі інструментів Meraki; «Приозерний».

Network type – який тип пристрою Meraki буде управлятися в цій мережі. При додаванні двох або більше типів пристроїв в мережу буде створена як комбінована мережа.

Add devices – якщо є номер замовлення або серійний номер пристрою, його можна додати тут.

4.2 Побудова картографічних карт всередині приміщень та імпорт карти в Meraki

Людам часто важко знайти дорогу в приміщенні, так як такі сервіси, як Google Maps, працюють тільки в торгових центрах і аеропортах. Хмара

Meraki надає місце розташування пристрою у вигляді пари широта/довгота. У більшості випадків вирівнювання планів поверхів будівлі на карті світу, виконане в Meraki, є приблизними і не виконується з урахуванням розташування в приміщенні, і, отже, не буде відповідати карті приміщень у додатку. З цієї причини потрібна додаткова обробка. Внутрішня картографічна платформа Mapwize дозволяє використовувати точки доступу Meraki для навігації по будівлям, таким як торгові центри, лікарні, фітес-центри і багато іншого.

Mapwize є платформою Indoor Mapping для створення нових цифрових сервісів всередині будівель.

Використовуючи інфраструктуру Indoor Location з відкритим вихідним кодом, Mapwize надзвичайно легко інтегрувати з Meraki Location API, щоб забезпечити синю точку на карті, відстеження активів і аналіз місця розташування. Детальний керівництво є на сайті create.meraki.io/build/wayfinding-mapwize.

Mapwize SDK доступні для iOS, Android і Javascript.

Після реєстрації на сайті mapwize.io ми отримаємо 30-денний безкоштовний доступ до Mapwize Studio, який дозволяє створити картографічну карту для однієї організації. Створимо карту для фітнес-центру «Приозерний» в м. Дніпро та додаймо місця згідно плану.

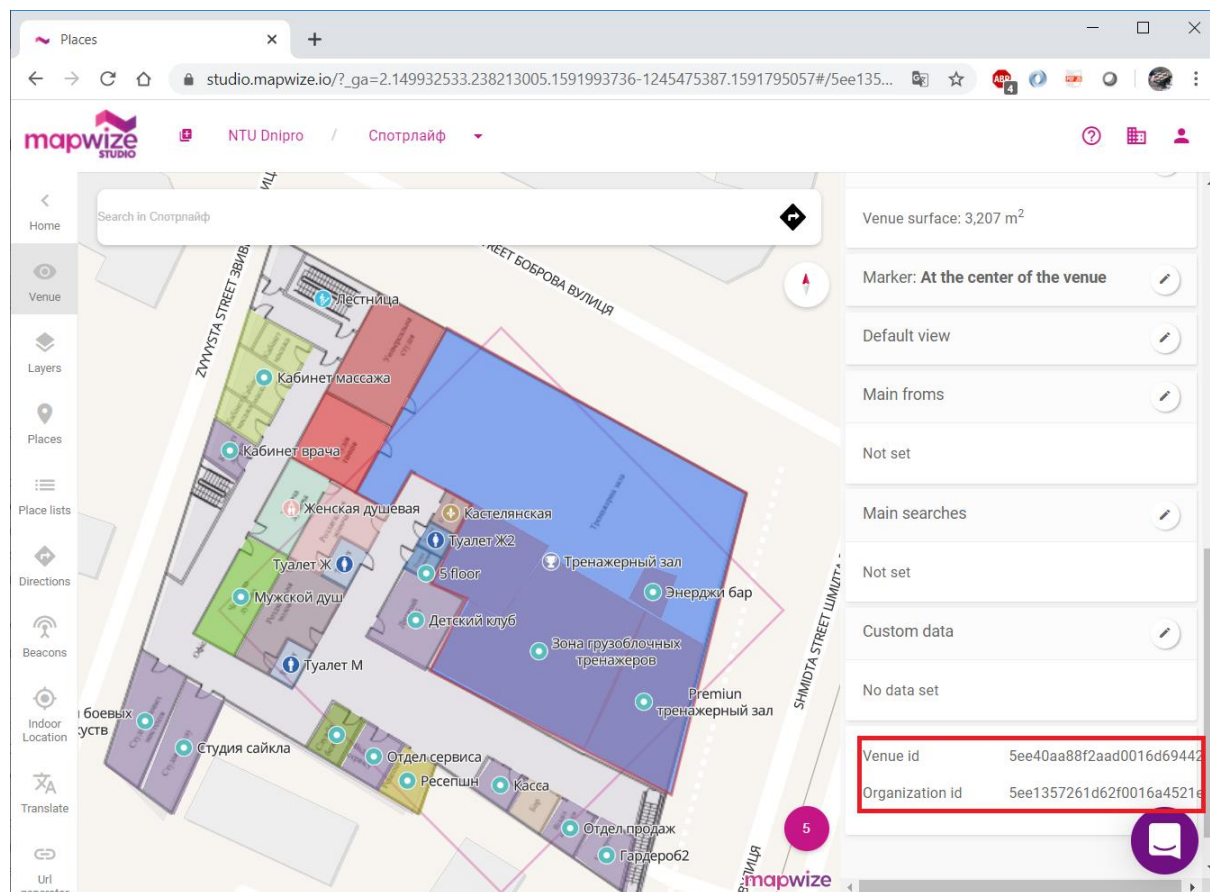
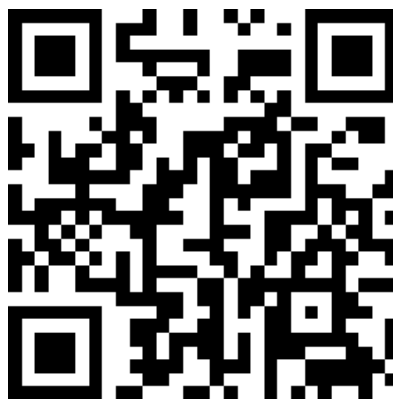


Рисунок 4.3 – Картографічна карта в Mapwize фітнес-центру «Приозерний»

До створеної карти можна перейти за посиланням https://maps.mapwize.io/#/v/_2d6f9222?z=15



Для створення запитів до карти з нових сервісів або додатків необхідні ідентифікатори Venue id та Organization id (рис. 3.9).

Після додавання пристроїв до Meraki Dashboard необхідно отримати поверхові плани Meraki та створити файл конфігурації, який буде використовуватися для обробки кожного місця розташування на карті Mapwize.

4.3 Отримання даних від Meraki

4.4 Отримання поверхових планів Meraki

В меню «Network-wide» панелі інструментів Meraki перейти до «Map & floor plans». Відкрити в браузері в режим розробника та знайти у вихідному коді параметр веб-сторінки Mkiconf.ng_id (рис. 3.10). Створити URL та отримати відповідь в форматі JSON:

https://api.meraki.com/node_groups/706502191543808183/floorplans

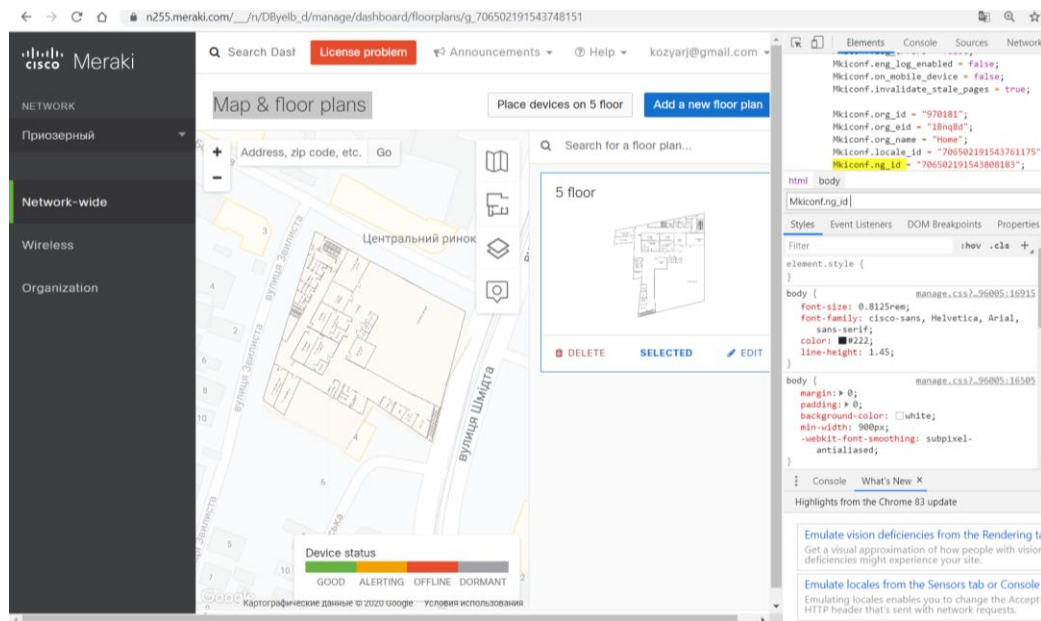


Рисунок 4.4 – Параметр веб-сторінки Mkiconf.ng_id

```
[
  {
    id: "g_706502191543748151",
    name: "5 floor",
    image_extension: "png",
    image_md5: "70d12a5080782387d103cc2b46ab29c3",
    image_url: "https://meraki-eu-central-1.s3.eu-central-1.amazonaws.com/assets/970181_70d12a5080782387d103cc2b46ab29c3.png?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIJNRMVZ3ANGOXGYA%2F20200614%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Date=20200614T003748Z&X-Amz-Expires=604800&X-Amz-SignedHeaders=host&X-Amz-Signature=a5f143721d3981cbea254fbc11155e9866ed1975a6070bbc32a9d67fa7c8cfd3",
    node_group_ids: [ ],
    locale_id: "706502191543761175",
    is_geoaligned: true,
    ne_lat: 48.4711803508711,
    ne_lng: 35.02242654562,
    node_ids: "",
    rotation_angle: -61.6706201999821,
    sw_lat: 48.4703481016623,
    sw_lng: 35.0213295221329
  }
]
```

4.5 Отримання списку точок доступу

Увійти в панель Meraki Dashboard. Перейти в меню Wireless->Access points. Справа вгорі натиснути кнопку Download As, щоб завантажити список точок доступу в форматі XML.

```
<?xml version="1.0" encoding="UTF-8"?>
- <network name="Приозерный - wireless">
  - <access_point is_gateway="false" id="3324006">
    <status>Has never connected to the Meraki cloud</status>
    <last_active>over 50 years</last_active>
    <name>00:18:0a:32:b8:66</name>
    <mac>00:18:0a:32:b8:66</mac>
    <serial>Q2DD-28FA-H7V5</serial>
    <model>MR16</model>
    <product_description>Meraki MR16 Cloud Managed AP</product_description>
    <lat>48.47085847833</lat>
    <lng>35.0217653810978</lng>
    <internet_address/>
    <usage type="bytes">0</usage>
    <client_count>0</client_count>
    <gateway_node>00:18:0a:00:00:00</gateway_node>
    <gateway_route_length>99</gateway_route_length>
    <last_active_timestamp type="timestamp">0</last_active_timestamp>
  </access_point>
</network>
```

4.6 Створення конфігуратора

В додатку А текст програми конфігуратора. Конфігуратор - це інструмент командного рядка, розроблений в NodeJS. Перед запуском необхідно встановити Node на ПК.

Запускаємо PowerShell і вказуємо шлях до програми.

```
cd H:\Install\Mapwize\meraki-indoor-location-master
```

```
H:\Install\Mapwize\meraki-indoor-location-master> npm install
commander --save
```

Використовуємо команду, щоб згенерувати файл конфігурації JSON, який пізніше буде використовуватися сервером.

```
./meraki-configurator/merakiFloorplansToMapwize.js --merakiFloorPlansConfig
[h:\API\wireless_devices_20200612_1806_to_20200613_1806.xml] --mapwizeUser
[panforova.ya.v@nmu.one] --mapwizePwd [w5s!EYCDQ5hJnUj] --mapwizeApiKey
[762f172c56468966217b1577daa33d38] --mapwizeOrganizationId
[5ee1357261d62f0016a4521e] --mapwizeVenueId [5ee40aa88f2aad0016d69442]
```

На цьому етапі створюються маяки в Meraki на основі конфігурації Meraki. Маяки будуть додані в Meraki на основі конфігурації плану поверху, виконаної в Meraki.

```
./meraki-configurator/configureFromMapwize.js --merakiFloorPlansConfig [h:\API\wireless_devices_20200612_1806_to_20200613_1806.xml] --mapwizeUser [panforova.ya.v@nmu.one] --mapwizePwd [w5s!EYCDQ5hJnUj] --mapwizeApiKey [4ca88cedc8593716] --mapwizeOrganizationId [5ee1357261d62f0016a4521e] --mapwizeVenueId [5ee40aa88f2aad0016d69442] --output [h:\API]
```

4.7 Використання API-інтерфейсу Cisco Meraki Dashboard

Для доступу до API спочатку необхідно включити API для організації в розділі Organization > Settings > Dashboard API access.

Dashboard API access

API Access ⓘ

Enable access to the Cisco Meraki Dashboard API

Після включення API необхідно перейти на сторінку свого профілю, щоб згенерувати ключ API. Ключ API пов'язаний з обліковим записом адміністратора Dashboard.

API access

API key

Generate API key

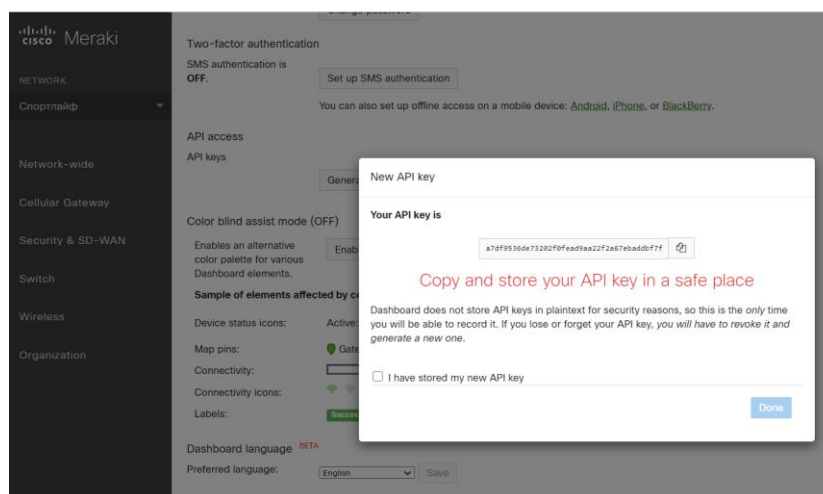


Рисунок 4.5 – Ключ API Meraki Dashboard
(a7df9536de73202f0fead9aa22f2a67ebaddbf7f)

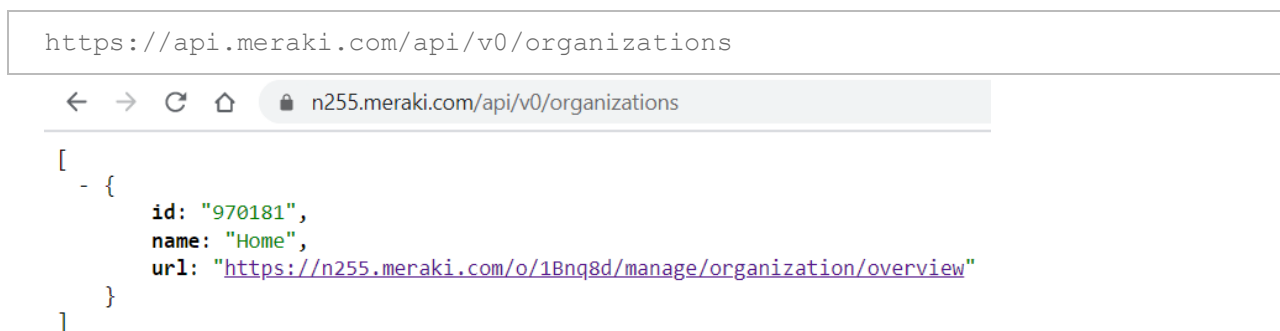
Кожен API-запит повинен вказувати ключ API в заголовку. Крім того, ключ API повинен бути зазначений в URL. API поверне 404 (а не 403) у відповідь на запит з відсутнім або неправильним ключем API, щоб запобігти витоку інформації про ресурси неавторизованих користувачам.

Версія API повинна бути вказана в URL:

`https://api.meraki.com/api/v0/ <ресурс>`

Наприклад:

```
https://api.meraki.com/api/v0/organizations
```



```
[
  - {
    id: "970181",
    name: "Home",
    url: "https://n255.meraki.com/o/1Bnq8d/manage/organization/overview"
  }
]
```

Отримати відомості про мережі:



```
[
  - {
    id: "L_706502191543761175",
    organizationId: "970181",
    name: "Приозерный",
    timeZone: "America/Los_Angeles",
    tags: null,
    - productTypes: [
      "appliance",
      "cellular gateway",
      "switch",
      "wireless"
    ],
    type: "combined",
    disableMyMerakiCom: false,
    disableRemoteStatusPage: true
  }
]
```

4.8 Побудова моделі в Packet Tracer та перевірка її роботи

Для дослідження Meraki на моделі побудуємо мережу в Packet Tracer. Функціонал програми на даний час представлений сервером Meraki та Secure Appliance meraki mx-65x (рис. 4.6). Схема адресації мереж та пристроїв завдана відповідно в табл. 4.1-4.2.

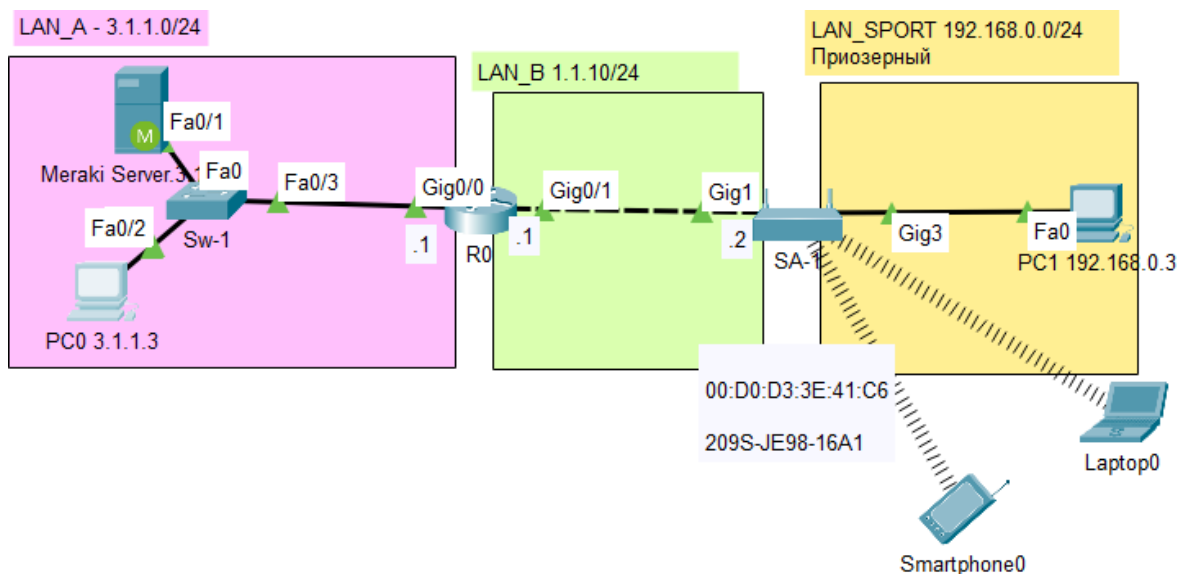


Рисунок 4.6 – Модель в Packet Tracer

Таблиця 2.1 – Схема адресації мереж

Мережа	IP-адреса	маска	Діапазон допустимих адрес
LAN_SPORT	3.1.1.0	255.255.255.0	1.1.1.1-1.1.1.254
LAN_INTERNET	1.1.1.0	255.255.255.0	3.1.1.1-3.1.1.254
LAN_VTY	192.168.0.0	255.255.255.0	192.168.1.1-192.168.0.254

Таблиця 2.2 – Таблиця адресації пристроїв

Мережа/Пристрій	Інтерфейс	IP-адреса	маска	ШЛЮЗ
LAN_SPORT	-	3.1.1.0	255.255.255.0	
LAN_INTERNET	-	1.1.1.0	255.255.255.0	
LAN_VTY		192.168.0.0	255.255.255.0	
R1	g0/0	3.1.1.1	255.255.255.0	
	g0/1	1.1.1.1	255.255.255.0	
SA-1	internet 1	3.1.1.2	255.255.255.0	
	vlan1	1.1.1.1	255.255.255.0	
Meraki Server	NIC	1.1.1.2	255.255.255.0	1.1.1.1
PC0	NIC	1.1.1.3	255.255.255.0	1.1.1.1
PC1	NIC	192.168.0.3	255.255.255.0	192.168.0.1

Розглянемо порядок налаштування Secure Appliance. Для цього потрібно зайти на Web-сторінку конфігурації з локальної мережі. За

замовчуванням для входу використовується серійний номер в якості логіна без пароля.

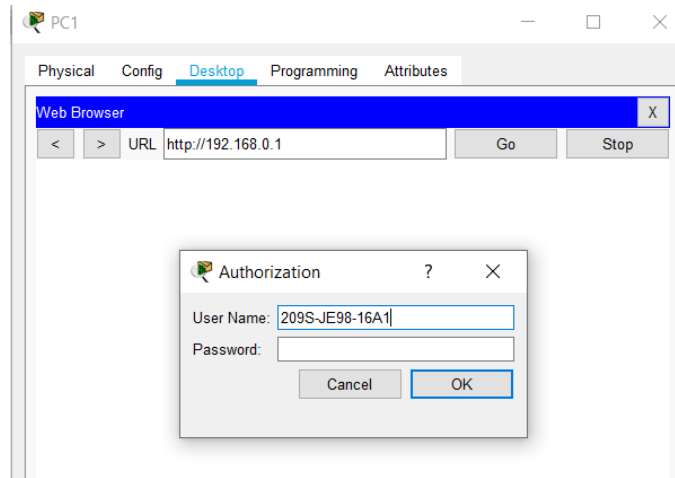


Рисунок 4.7 – Веб-авторизація на Secure Appliance

На вкладці Configure налаштовуються параметри IP-адресації інтерфейсу підключення до інтернету. Для збереження налаштувань використовується кнопка Save.

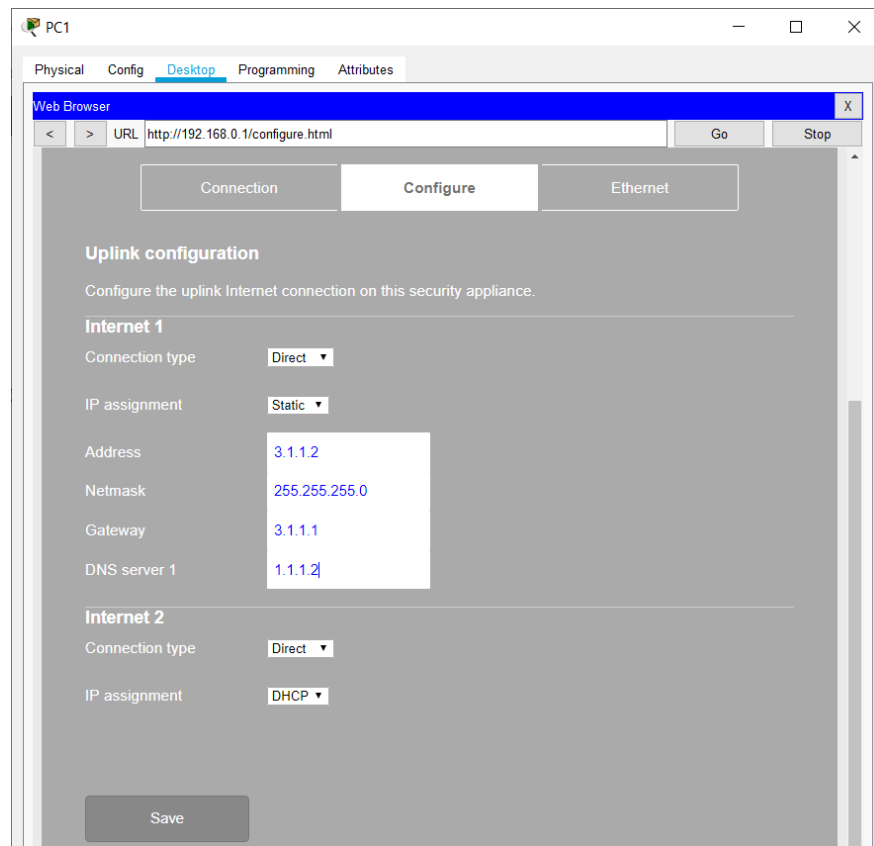


Рисунок 4.8 – Налаштування IP-адресації до Інтернет

Розглянемо налаштування Meraki Server. Для переходу на сторінку налаштувань використовується посилання <https://dashboard.meraki.com>, далі потрібно створити користувача для адміністрування (наприклад cisco@gmail.com, password cisco12345).

Для створення мережі потрібно перейти на вкладку Create a network, ввести назву нової мережі та натиснути кнопку Create network (рис. 4.9). В цьому ж вікні можна додати новий пристрій в хмару.

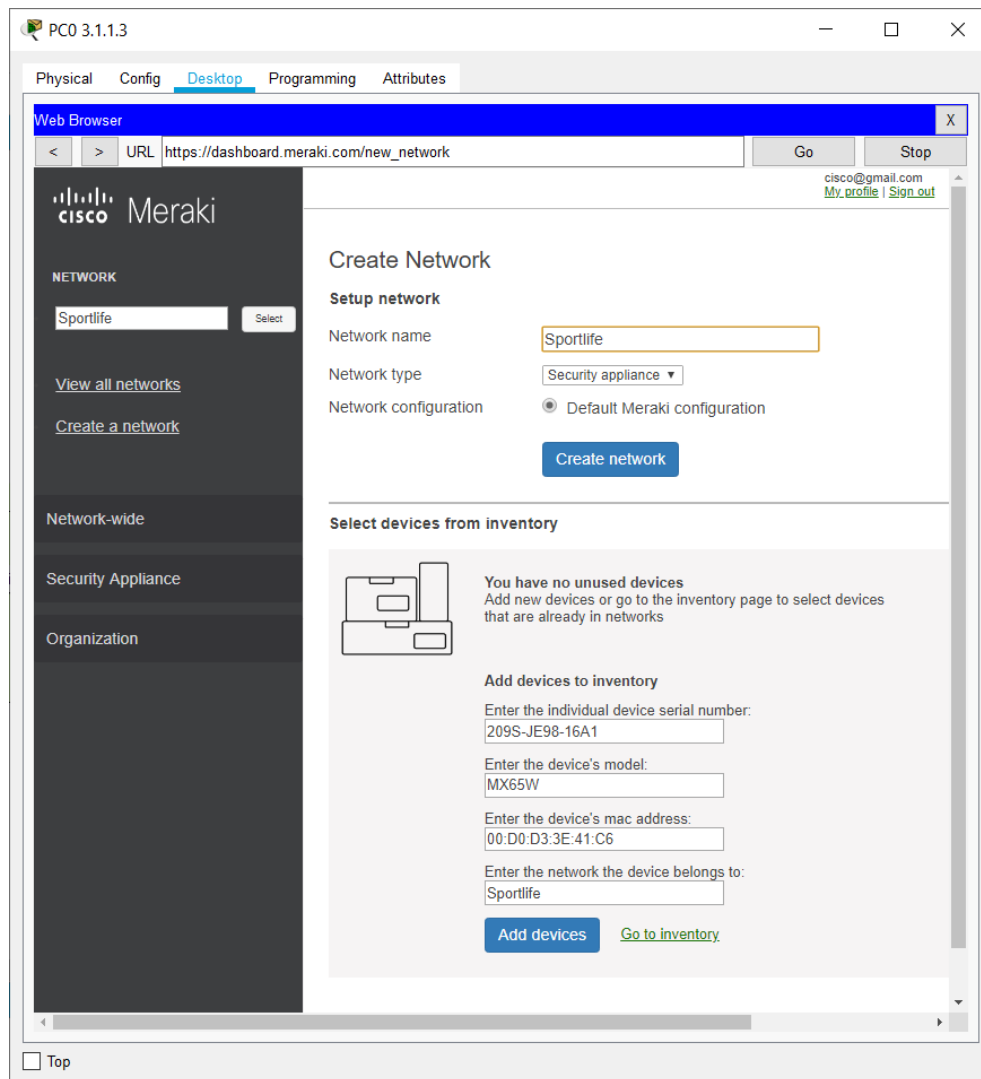


Рисунок 4.9 – Додавання мережі та пристрою в панель Meraki Dashboard

Для налаштування параметрів бездротової мережі потрібно на вкладці Security Appliance перейти на вкладку Wireless settings і ввести параметри SSID, захисту з'єднання та натиснути кнопку Save Changes.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Техніко-економічне обґрунтування розробки

В кваліфікаційній роботі розглядається удосконалення комп'ютерної системи фітнес-клубу «Спортлайф» з опрацюванням побудови, налаштування та безпеки корпоративної мережі. Для удосконалення КС необхідно облаштувати фітнес-клуб бездротовими точками доступу Meraki MR32, підвищити ефективність налаштування та керування існуючими та новими мережами філіалів, збір інформації в режимі реального часу з моніторингу та геопозиціонування, тепловими картками клієнтів, відвідуваності різних зон, і т.д.

Для обґрунтування економічної доцільності застосування КС, необхідно виконати:

- розрахунок капітальних витрат на придбання складових КС;
- розрахунок річних експлуатаційних витрат проектної апаратури;
- величину річного економічного ефекту.

5.2 Розрахунок капітальних витрат на придбання складових КС

Капітальні вкладення – це кошти, призначені для створення і придбання основних фондів та нематеріальних активів, що підлягають амортизації.

Кошторис капітальних витрат на обладнання, яке необхідно для реалізації комп'ютерної системи, приведена в таблиці 5.1.

Капітальні витрати розраховуються за формулою:

$$K_{\text{пр}} = K_{\text{об}} + K_{\text{тр}} + K_{\text{мн}} + K_{\text{пз}}, \quad (5.1)$$

де $K_{\text{об}}$ – вартість обладнання, грн.,

$K_{\text{тр}}$ – вартість транспортно-заготівельних витрат, грн.,

$K_{\text{мн}}$ – вартість монтажних-налагоджувальних робіт, грн.,

$K_{\text{пз}}$ – вартість розробки програмного забезпечення.

Таблиця 3.1 – Кошторис капітальних витрат

№ п/п	Найменування обладнання	Ед. виміру	Кількість	Вартість од. облад-я, грн	Сумма, грн.
1	Маршрутизатор Wi-Fi Meraki MR32	шт	6	21 484	128904
Загалом					128904

Загальна вартість обладнання $K_{об}=128904$ грн.

Вартість транспортно-заготівельних і складських витрат становить 7% від вартості обладнання.

$K_{тр}=128904 * 7\% = 9023,28$ грн.

Вартість монтажних-налагоджувальних робіт становить 8% від вартості обладнання.

$K_{мн}=128904 * 8\% = 10312,32$ грн.

Проектні капітальні витрати на обладнання складуть:

$$K_{пр.об} = 128904 + 9023,28 + 10312,32 = 148239,6 \text{ грн}$$

5.3 Розрахунок капітальних витрат на програмне забезпечення

5.3.1 Розрахунок часу на розробку програмного забезпечення

Трудомісткість розробки програмного забезпечення:

$$t = t_o + t_d + t_a + t_n + t_{нал} + t_{док}, \quad (5.2)$$

где t_o - витрати праці на підготовку й опис поставленого завдання

t_d - витрати праці на дослідження алгоритму розв'язку завдання;

t_a - витрати праці на обробку блок-схеми алгоритму;

t_n - витрати праці на програмування по готовій блок-схемі;

$t_{нал}$ - витрати праці на налаштування програм на ЕОМ;

$t_{док}$ - витрати праці на підготовку документації за завданням.

Складові частини витрат праці визначаються на підставі умовної кількості оброблюваних операторів у програмному забезпеченні. До них відносять ті оператори, які необхідно написати в процесі роботи над

програмою з урахуванням можливих уточнень у постановці завдання й удосконалення алгоритму.

Умовна кількість операторів у програмі:

$$Q = q \cdot c \cdot (1+p) \quad (5.3)$$

де q – кількість операторів, використовуваних у програмі.

c – коефіцієнт складності програми;

p – коефіцієнт корекції програми в процесі її обробки.

Приймаємо $q = 400$.

Коефіцієнт складності «с» програми визначає відносну складність програми відносно типового завдання, складність якого відповідає 1. Приймаємо $c = 1,25$.

Коефіцієнт корекції програми «р» визначає збільшення обсягу робіт за рахунок внесення змін в алгоритм або програму в результаті уточнення постановки завдання. Ухвалюємо $p=0,1$, це відповідає внесенню 3...5 корекцій, що тягнуть за собою переробку 5-10% готової програми.

Таким чином, для програми, описаної в кваліфікаційній роботі:

$$Q = 400 \cdot 1,25(1+0,1) = 550$$

Оцінка витрат праці на підготовку й опис завдання становлять

$$t_0 = 50 \text{ люд.-годин.}$$

Витрати праці на вивчення опису завдання визначаються з урахуванням уточнення опису й кваліфікації програміста по формулі:

$$t_0 = \frac{Q \cdot B}{(75 \dots 85) \cdot k} \text{ люд.-годин} \quad (5.4)$$

де B – коефіцієнт збільшення витрат праці, $B=1,4$;

k – коефіцієнт кваліфікації програміста, які визначається залежно від стажу роботи зі спеціальності. У нашому випадку коефіцієнт кваліфікації програміста становить $k=1,2$.

Для розроблюваного програмного забезпечення:

$$t_D = \frac{550 \cdot 1,4}{80 \cdot 1,2} = 8,59 \text{ люд.-годин.}$$

Витрати на розробку алгоритму розв'язку завдання:

$$t_a = \frac{Q}{(20...25) \cdot k} \text{ люд.-годин} \quad (5.5)$$

Для розроблювального програмного забезпечення:

$$t_a = \frac{550}{20 \cdot 1,2} = 22,92 \text{ люд.-годин.}$$

Витрати праці на складання програми по готовій блок-схемі алгоритму:

$$t_n = \frac{Q}{(20...25) \cdot k} \text{ люд.-годин} \quad (5.6)$$

Для розроблюваного програмного продукту:

$$t_n = \frac{550}{20 \cdot 1,2} = 22,92 \text{ люд.-годин.}$$

Витрати праці на налагодження програми на ЕОМ розраховуються по формулі:

$$t_{нал} = \frac{Q}{(4...5) \cdot k} \text{ люд.-годин} \quad (5.7)$$

Для конкретного програмного продукту:

$$t_{нал} = \frac{550}{5 \cdot 1,2} = 22,92 \text{ люд.-годин.}$$

Витрати праці на підготовку документації за завданням визначаються по формулі:

$$t_d = t_{др} + t_{до}, \text{ люд.-година} \quad (5.8)$$

де $t_{др}$ – трудомісткість підготовки матеріалів до написання;

$t_{до}$ – трудомісткість редагування, друку й оформлення документації.

$$t_{др} = Q / (15...20) \cdot k, \quad (5.9)$$

$$t_{др} = 550 / 18 \cdot 1,2 = 25,47 \text{ люд.-година;}$$

$$t_{до} = 0,75 \cdot t_{др}, \quad (5.10)$$

$$t_{до} = 0,75 \cdot 25,47 = 19,10 \text{ люд.-година.}$$

Для розроблюваного програмного забезпечення витрати праці на підготовку документації за завданням будуть становити:

$t_D = 6,37 + 4,77 = 44,56$ люд.-година.

Трудовістіть розробки програмного забезпечення буде становити:

$t = 50 + 2,15 + 5,73 + 5,73 + 22,92 + 11,14 = 240,65$ людино-годин.

5.3.2 Розрахунки витрат на розробку програмного продукту

Витрати на розробку програмного продукту $K_{ПЗ}$ містять витрати на заробітну плату розробника програми $Z_{ЗП}$ і вартість машинного часу, необхідного для налаштування програми на ЕОМ $Z_{МЧ}$

$$K_{ПЗ} = Z_{ЗП} + Z_{МЧ}, \text{ грн.} \quad (5.11)$$

Заробітна плата розробника програмного забезпечення:

$$Z_{ЗП} = t \cdot C_{ПР}, \text{ грн.} \quad (5.12)$$

де t – загальна трудовістіть обробки програмного забезпечення;

$C_{ПР}$ – середня годинна тарифна ставка програміста становить:

$C_{ПР} = 155$ грн./година.

Заробітна плата за розробку програмного забезпечення дорівнює:

$Z_{ЗП} = 240,65 \cdot 155 = 37301,36$ грн.

Вартість машинного часу, необхідного для налаштування програми на ЕОМ:

$$Z_{МЧ} = t_{НАЛ} \cdot C_{МЧ}, \text{ грн.} \quad (5.13)$$

де:

$t_{НАЛ}$ – трудовістіть налаштування програми на ЕОМ, людино-годин;

$C_{МЧ}$ – вартість машино-години ЕОМ, грн./година. $C_{МЧ} = 38,04$ грн./година.

Вартість 1 години машинного часу ПК визначається за формулою:

$Z_{МЧ} = 22,92 \cdot 38,04 = 854,33$ грн.

Вартість 1 години машинного часу ПК визначається за формулою

$$C_{МЧ} = P \cdot t_{НАЛ} \cdot C_e + \frac{\Phi_{Зал} \cdot H_a}{F_p} + \frac{K_{ЛПЗ} \cdot H_{ЛПЗ}}{F_p} \quad (5.14)$$

$C_{МЧ} = (0,6 \cdot 22,92 \cdot 0,642) + (3000 \cdot 0,5) / 1920 + (15000 \cdot 0,25) / 1920 = 38,04$

грн/год

де $P=0,6$ – встановлена потужність ПК, кВт;

$C_e=0,642$ – тариф на електричну енергію з ПДВ, грн/кВт*година;

(Тариф відповідно до тарифів ПАТ «ДТЕК Дніпрообленерго для споживачів 2 класу з 01 січня 2020 року» = 535,6 грн/МВт без ПДВ»)

$\Phi_{зал}=3000$ – залишкова вартість ПК на поточний рік, грн.;

$H_a=0,5$ – річна норма амортизації на ПК, частки одиниці;

$H_{анз}=0,25$ – річна норма амортизації на ліцензійне програмне забезпечення, частки одиниці;

$K_{анз}=15000$ грн, вартість ліцензійного програмного забезпечення, грн.(табл.4.2.);

$F_p=1920$ – річний фонд робочого часу (за 40-годинного робочого тижня $F_p=1920$).

Таблиця 5.1 – Вартість необхідного програмного забезпечення

Програмне забезпечення	Вартість, грн
Meraki Dashboard	0
Підписка studio.marwize.io	15000
Cisco Packet Tracer	0
Всього	15000

Витрати на розробку програмного забезпечення системи керування будуть становити:

$$K_{пз} = 37301,36 + 854,33 = 38155,69 \text{ грн.}$$

Певні, таким чином, витрати на створення програмного забезпечення є частиною одноразових капітальних витрат на створення системи керування.

Очікувана тривалість розробки програмного забезпечення:

$$T = \frac{t}{B_k \cdot F_p}, \text{ міс.} \quad (5.15)$$

де, B_k – кількість розробників. Програма розроблялася однією людиною, тому $B_k=1$;

F_p – місячний фонд робочого часу ($F_p = 176$ годин).

Визначимо тривалість розробки ПО:

$$T = \frac{240,65}{1 \cdot 176} = 1,37 \text{ міс.}$$

Таким чином, капітальні витрати розраховані за формулою (5.1) дорівнюють:

$$K_{\text{пр}} = 128904 + 9023,28 + 10312,32 + 40788,76 = 189028,36 \text{ грн.}$$

Висновок

При розробці компютерної мережі капітальні витрати 189028,36 грн, у тому числі капітальні витрати на обладнання мережі 128904 грн та витрати на оплату праці по розробці моделі компютерної мережі 40788,76 грн.

У загальній сумі витрат на розробку мережі.

Вартість комплектуючих складає – 68,2 %.

Витрати на монтаж-налагоджувані та транспортні роботи – 10%.

Заробітна плата на розробку моделі – 20%.

Витрати на використання ЕОМ – 1,8%.

Всього: 100%.

Найбільша частка витрат – витрати на комплектуючі – 68,2 %.

6 ОХОРОНА ПРАЦІ

При виконанні кваліфікаційної роботи необхідно дотримуватись вимог охорони праці, передбачених тематикою проекту. В даній роботі це правила для програміста ПК.

Перелік нормативно-правових актів, які регулюють це питання, досить широкий. Наприклад, ст. 21 Кодексу законів про працю України визначає обов'язки роботодавця щодо забезпечення працівникам комфортних та безпечних умов праці, а ст.13 Закону України «Про охорону праці» закріплює це право з позиції охорони праці. Основні нормативні акти, що врегульовують окремі моменти щодо техніки безпеки та обладнання:

- Наказ Держгірпромнагляду України «Про затвердження Правил охорони праці під час експлуатації електронно-обчислювальних машин» від 26 березня 2010 р. № 65;
- Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПіН 3.3.2.007-98, затверджені постановою Головного державного санітарного лікаря України від 10 грудня 1998 р. № 7;
- Примірні інструкція з охорони праці під час експлуатації електронно-обчислювальних машин, затверджена наказом Міністерства доходів і зборів України від 5 вересня 2013 р. № 443.

6.1 Загальні положення

До роботи на персональному комп'ютері допускають осіб, які пройшли інструктаж з питань охорони праці та пожежної безпеки.

Користувач зобов'язаний:

- виконувати правила внутрішнього трудового розпорядку;
- не допускати за своє робоче місце сторонніх осіб;

- не виконувати вказівок, які суперечать правилам охорони праці та пожежної безпеки;
- знати правила надання домедичної допомоги;
- знати розташування та вміти користуватись первинними засобами пожежогасіння;
- вміти працювати з ПК.

Основні небезпечні та шкідливі виробничі фактори, що можуть впливати на користувача:

- підвищений рівень статичної електрики;
- нерівномірність розподілу яскравості в полі зору;
- підвищена яскравість світлового зображення;
- ураження електричним струмом;
- напруга зору та уваги;
- тривалі статичні навантаження.

Мережевий адміністратор проходить позаплановий інструктаж: при зміні технологічного процесу або правил по охороні праці, заміні або модернізації устаткування, зміні умов і організації праці.

Мережевий адміністратор зобов'язаний:

- дотримуватися правил внутрішнього трудового розпорядку;
- дотримуватися правил особистої гігієни, перед прийомом їжі необхідно мити руки з милом.

Палити дозволяється тільки в спеціально відведеному та обладнаному для цього місці, приймати їжу дозволяється в кімнаті відпочинку і прийому їжі.

У приміщеннях із ПК має бути природне і штучне освітлення.

При розміщенні робочих місць необхідно унеможливити пряме засвічування екрана природним освітленням.

При природному освітленні слід передбачити наявність сонцезахисних засобів (плівка, жалюзі, штори тощо).

Світлові відблиски із клавіатури, екрана та інших частин ПК у напрямку очей користувача неприпустимі.

Основним обладнанням робочого місця є ПК або ноутбук, монітор, клавіатура, маніпулятор, робочий стіл, стілець (крісло).

При розміщенні елементів робочого місця слід враховувати:

- робочу позу користувача;
- простір для розміщення користувача;
- можливість огляду елементів робочого місця;
- можливість огляду простору поза межами робочого місця;
- можливість робити записи, розміщувати на робочому столі документацію та матеріали, які використовує користувач.

Розміщення елементів робочого місця не має заважати рухам та переміщенню для експлуатування ПК.

Монітор встановлюють так, щоб відстань від поверхні екрана до очей користувача була 600-700 мм залежно від розміру екрана.

Клавіатуру розміщують на робочому або окремому столі на відстані 100-300 мм від краю з боку користувача. Положення клавіатури та кут її нахилу залежить від побажання користувача (як правило, в межах 5-15°). Не допускати хитання клавіатури.

Конструкція робочого столу має бути такою, щоб оптимально розмістити на робочій поверхні обладнання, що використовують, з урахуванням кількості, розмірів, конструктивних особливостей і характеру його роботи.

Крісло має забезпечувати підтримування раціональної робочої пози під час виконання основних виробничих операцій та можливість зміни пози. Тип робочого крісла обирають залежно від характеру та тривалості роботи.

Раціональна поза користувача:

- ступні розташовані на підлозі або на підставці для ніг;
- стегна зорієнтовані у горизонтальній площині;
- верхні ділянки рук вертикальні;

- кут ліктьового суглоба у межах 70-90°;
- зап'ястя зігнуті під кутом не більше ніж 20°;
- нахил голови у межах 15-20°, а часті її повороти виключені.

Для забезпечення оптимальної робочої пози користувача необхідно:

- засоби праці, з якими користувач має тривалий або найбільш частий зоровий контакт, розмістити у центрі зони зорового спостереження та моніторного поля;
- забезпечити відстань близько 500 мм між найважливішими засобами праці, з якими користувач працює найчастіше.

ПК встановлювати на рівній твердій поверхні (столі). Не дозволено встановлювати ПК та оргтехніку на хитких підставках чи на похилій поверхні.

ПК не встановлювати впритул до стіни, перегородки тощо. Не допускати загородження вентиляційних отворів ПК сторонніми предметами.

Розетка біля ПК має бути в доступному місці, щоб в аварійних випадках можна було своєчасно його відімкнути. Не рекомендовано використовувати подовжувачі.

Під час переміщення ПК, периферійних пристроїв витягти вилку живлення з розетки.

Не допускати ушкодження чи модифікування шнура живлення. Заборонено ставити важкі речі на шнур живлення, тягнути чи надмірно перегинати його, скручувати та перев'язувати шнур живлення вузлом.

ПК під'єднувати до електромережі лише за допомогою справних штепсельних з'єднань та електричних розеток заводського виробництва.

Штепсельні з'єднання та електричні розетки мають бути зі спеціальними контактами для під'єднання нульового захисного провідника. Їхня конструкція має забезпечувати з'єднання нульового захисного провідника раніше, ніж з'єднання фазового та нульового робочого провідників. Порядок роз'єднань при вимкненні має бути зворотнім.

Заборонено під'єднувати електрообладнання до звичайної двошнурової електромережі.

За невиконання цієї інструкції працівники несуть відповідальність згідно з чинним законодавством.

6.2 Вимоги безпеки перед початком роботи на ПК

Оглянути робоче місце і навести на ньому лад; впевнитись, що на ньому немає сторонніх предмети, все обладнання і блоки ПК з'єднані з системним блоком з'єднувальними шнурами.

Перевірити надійність встановлення апаратури на робочому столі. Монітор не має стояти на краю стола. Повернути монітор так, щоб було зручно дивитися на екран — під прямим кутом (а не збоку) і трохи зверху вниз; при цьому екран має бути трохи нахиленим — нижній край ближче до користувача.

Перевірити загальний стан апаратури, справність електропроводки, з'єднувальних шнурів, штепсельних вилок, розеток, заземлення захисного екрана.

Вставити вилку в розетку і впевнитися, що вона міцно тримається. Заборонено вставляти і виймати вилку мокрими руками.

Відрегулювати та зафіксувати висоту крісла та зручний для користувача нахил спинки.

За потреби приєднати до комп'ютера необхідну апаратуру (принтер, сканер тощо). Усі кабелі, що з'єднують системний блок із іншими пристроями, вмикати та вимикати лише при вимкненому комп'ютері.

Відрегулювати яскравість свічення, контрастність монітора.

Про всі виявлені несправності інформувати керівника робіт і не братися до роботи, доки їх не буде усунено.

6.3 Вимоги безпеки під час виконання роботи на ПК

Під час роботи на ПК:

- стійко встановити клавіатуру на робочому столі, не допускаючи її хитання, водночас передбачити можливість її поворотів та переміщень;
- якщо в конструкції клавіатури не передбачено простору для упору долонь, клавіатуру розміщують на відстані не менше 100 мм від краю столу в оптимальній зоні моніторного поля;
- під час роботи на клавіатурі сидіти рівно, не напружуватися;
- щоб зменшити несприятливе навантаження на користувача при роботі з комп'ютерною мишею (вимушена поза, необхідність постійно контролювати якість дій), забезпечити велику вільну поверхню столу для переміщення комп'ютерної миші та зручного упору ліктьового суглоба;
- періодично при вимкненому комп'ютері прибирати пил із поверхонь апаратури спеціальними серветками.

При роботі з ПК заборонено:

- самостійно розбирати та ремонтувати системний блок (корпус ноутбука), монітор, клавіатуру, комп'ютерну мишу тощо;
- встромляти сторонні предмети до вентиляційних отворів ПК, ноутбука або монітора;
- ставити на системний блок ПК та периферійні пристрої металеві предмети, ємкості з водою (вази, горщики для квітів, склянки), оскільки через потрапляння води у середину апарата може виникнути пожежа або ураження електрострумом.

Тривалість безперервної роботи за ПК не має перевищувати 2 год.

Після цього необхідно зробити 15-хвилинну перерву.

Якщо виник зоровий дискомфорт або інші неприємні відчуття, необхідно зробити коротку перерву.

Для зниження нервово-емоційного напруження, стомлення зорового аналізатора, поліпшення мозкового кровообігу, подолання несприятливих

наслідків гіподинамії, запобігання втомі доцільно під час декількох перерв виконувати комплекс вправ.

6.4 Вимоги безпеки після закінчення роботи на ПК

Зберегти інформацію.

Перевірити відсутність інструментів на вузлах ремонтowanego обладнання, зібрати і покласти їх у відведене місце.

Привести в порядок робоче місце, провести прибирання ділянки, на якій виконувалася робота.

Вимкнути ПК, монітор чи ноутбук.

Вимкнути стабілізатор, якщо комп'ютер під'єднаний до мережі через нього.

Прибрати робоче місце.

6.5 Вимоги безпеки в аварійній ситуації

Аварійні та небезпечні ситуації під час виконання роботи на ПК можуть виникнути у разі: короткого замикання, перевантаження блоку живлення системного блоку, перегрівання, пожежі, поломки крісла тощо.

У разі виникнення аварії або ситуації, що може привести до аварії, нещасного випадку, негайно від'єднати ПК від електромережі, повідомити інцидент керівникові.

Не допускати в небезпечну зону сторонніх осіб.

Якщо стався нещасний випадок, зберегти обстановку в робочій зоні та устаткування у такому стані, в якому вони були на момент події (якщо це не загрожує життю і здоров'ю інших працівників і не призведе до більш тяжких наслідків). Поінформувати про подію керівника робіт (іншу відповідальну особу підприємства) та в подальшому керуватися його вказівками. Вжити заходів, щоб запобігти подібним випадкам у подальшому.

У разі виникнення пожежі (ознак горіння), повідомити керівнику та, за потреби, викликати оперативно-рятувальну службу за телефоном 101 або 102 (назвати адресу та місце виникнення пожежі, наявність людей, повідомити своє прізвище) та вжити можливих заходів для евакуування людей, гасіння (локалізації) пожежі наявними засобами пожежогасіння. Пам'ятати, що гасіння електротехнічних пристроїв, які перебувають під напругою, виконувати лише після їх попереднього від'єднання від електромережі. Гасити за допомогою вуглекислотних або порошкових вогнегасників, а в окремих випадках — сухим піском.

За потреби надати потерпілому домедичну допомогу згідно з інструкцією, що діє на підприємстві. У разі подальшого погіршення самопочуття потерпілого, не припиняючи надання домедичної допомоги, викликати за телефоном 103 швидку медичну допомогу.

Виконувати вказівки керівника робіт для ліквідування небезпеки.

5.4 Безпека у випадку надзвичайної ситуації

На території Дніпропетровської області, у порівнянні з іншими регіонами, надзвичайні ситуації природного характеру, що спостерігаються нечасто. У регіоні практично не буває катастрофічних землетрусів, сходу сніжних лавин і зсувів, морози рідко досягають -30°C , а спека $+40^{\circ}\text{C}$.

При виявленні небезпечної ситуації (пожежа, землетрус, радіаційна безпека, неполадки в електрогосподарстві тощо) для власного життя та життя співробітників заспокоїти і заспокоїти оточуючих.

Не усувати самому несправностей електромережі та електрообладнання, а вимкнути загальне електропостачання.

При виявленні пожежі зобов'язаний негайно викликати пожежну частину.

Вжити заходів згідно з планом евакуації на випадок пожежі, виробничих та природних явищ та вивести працівників у безпечне місце. Організувати роботу ДПД щодо збереження майна та цінних паперів.

При появі сторонньої особи, яка застосовує протиправні дії щодо безпеки життєдіяльності оточуючих, викликати міліцію.

У випадку травмування працівників або клієнтів під час роботи підприємства необхідно викликати швидку допомогу або за потреби надати першу долікарську допомогу, за необхідності створити комісію по розслідування нещасного випадку, видати акт встановленого зразка, наказ про підсумки розслідування, повідомлення про наслідки нещасного випадку.

ВИСНОВКИ

У кваліфікаційній роботі розглянуто проектування корпоративної мережі фітнес клубу «Спортлайф» шляхом оновлення безпроводних точок доступу на моделі з підтримкою хмарної платформи Meraki.

Практична значимість роботи полягає в можливості впровадження результатів модернізації не тільки в даному в одному фітнес-центрі, а і в інших філіалах по Україні. Це дозволить не тільки централізовано з будь-якого місця керувати мережним обладнання через веб-браузерну панель Meraki Dashboard, а дозволить розробникам використовувати і розширювати можливості, сервіси та дані платформи в формі відкритих API-інтерфейсів.

Розробка комп'ютерної мережі виконана відповідно до завдання на кваліфікаційну роботу бакалавра.

Розроблена схема мережі реалізована у вигляді моделі на базі симулятора Cisco Packet Tracer і перевірена її робота. Був написаний програмний код для отримання коректної інформації з пристроїв.

Переваги проекту:

- масштабоване рішення;
- централізоване управління по всій мережі;
- інструменти віддаленого вирішення проблем;
- дуже прості і швидкі розгортання і автоматичні оновлення;
- зниження витрат на впровадження і обслуговування мережі;
- не потрібно апаратний або програмний контролер.

ПЕРЕЛІК ПОСИЛАНЬ

1. <https://developer.cisco.com> (дата звернення 19.06.2019)
2. <https://devnetsandbox.cisco.com> (дата звернення 19.06.2019)
3. <https://www.netacad.com> (дата звернення 19.06.2019)
4. <https://sandboxdnac.cisco.com> (дата звернення 19.06.2019)
5. (https://uk.wikipedia.org/wiki/Прикладний_програмний_інтерфейс)
6. https://documentation.meraki.com/Architectures_and_Best_Practices/MX_and_MS_Basic_Recommended_Layer_3_Topology
7. <https://comtrade.ua/cisco-meraki-mr34/> 37386
8. https://documentation.meraki.com/MR/Monitoring_and_Reporting/Location_Analytics
9. Межсетевые экраны нового поколения. [Електронний ресурс] – Режим доступу до ресурсу: <https://static-course-assets.s3.amazonaws.com/CCNAS2/ru/index.html#4.2.2.5> (20.05.2020)
10. Цвіркун Л.І. Комп'ютерні мережі. Методичні рекомендації до виконання курсового проекту студентами галузі знань 12 Інформаційні технології спеціальності 123 Комп'ютерна інженерія / Л.І. Цвіркун, Я.В. Панферова, Л.В. Бешта; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро: НТУ «ДП», 2018. – 28 с.
11. Цвіркун Л.І. Комп'ютерні мережі. Методичні рекомендації до виконання лабораторних робіт студентами галузі знань 12 Інформаційні технології спеціальності 123 Комп'ютерна інженерія: у 2 ч. / Л.І. Цвіркун, Я.В. Панферова ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро: НТУ «ДП», 2018. – Ч. 1. – 60 с.
12. Методичні вказівки з виконання економічного розділу в дипломних проектах студентів спеціальності “Комп'ютерні системи” / Уклад. О.Г. Вагонова, О.Б. Нікітіна Н.Н. Романюк – Д.: НГУ. – 2013. – 11 с.
13. Охорона праці. Методичні вказівки до виконання розділу „Охорона праці та безпека в надзвичайних ситуаціях“ в дипломних проектах

(роботах) студентів інституту електроенергетики / В.І. Голінько, В.Ю. Фрундін, М.Ю. Іконніков – Д.: НГУ, 2011. – 7 с.

14. Атестація здобувачів вищої освіти. 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», 2020. – 69 с.

Додаток А

JavaScript для імпорту даних з Meraki в Mapwize

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

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
НАЛАШТУВАННЯ МЕРЕЖІ КОМП'ЮТЕРНОЇ СИСТЕМИ

Текст програми

804.02070743.20005-01 12 01

Листів 7

2020

АНОТАЦІЯ

Дана програма містить код JavaScript для імпорту даних з Meraki в Mapwize і створення файлу конфігурації, необхідного для запуску сервера прослуховування Meraki.

3МІСТ

	Стор.
1. merakiFloorplansToMapwize.js	4
2. merakiAccessPointsToMapwize.js	8

```

1. merakiFloorplansToMapwize.js
!/usr/bin/env node

var _ = require('lodash');
var async = require('async');
var fs = require('fs');
var MapwizeAPI = require('mapwize-node-api');
var chalk = require('chalk');
var request = require('request');
var imageSize = require('image-size');
var program = require('commander');
var mapwize = require('./meraki-server/utils/mapwize');

program
  .version('0.1.0')
  .description('Upload Meraki data into Mapwize')
  .option('--merakiFloorPlansConfig [filepath]', 'Filepath to the Meraki
configuration for floor plans')
  .option('--mapwizeApiUrl [url]', 'Mapwize API URL')
  .option('--mapwizeUser [user]', 'Mapwize user mail')
  .option('--mapwizePwd [password]', 'Mapwize user password')
  .option('--mapwizeApiKey [key]', 'Mapwize api key')
  .option('--mapwizeOrganizationId [organizationId]', 'Mapwize
organizationId')
  .option('--mapwizeVenueId [venueId]', 'Mapwize venueId')
  .option('--mapwizeUniverseId [universeId]', 'Mapwize universeId')
  .option('--reset', 'Reset Meraki layers')
  .parse(process.argv)

  if (!program.merakiFloorPlansConfig || !program.mapwizeUser ||
!program.mapwizePwd || !program.mapwizeApiKey ||
!program.mapwizeOrganizationId || !program.mapwizeVenueId) {
    console.log(chalk.red('- The arguments --merakiFloorPlansConfig, --
mapwizeUser, --mapwizePwd, --mapwizeApiKey, --mapwizeOrganizationId, --
mapwizeVenueId are required.'));
    return;
  }

  if (!program.mapwizeApiUrl) {
    program.mapwizeApiUrl = 'https://api.mapwize.io';
  }

  //Mapwize client
  var MapwizeClient = new MapwizeAPI(program.mapwizeApiKey,
program.mapwizeOrganizationId, {serverUrl: program.mapwizeApiUrl});

```

```

//Meraki floorplans JSON
var merakiFloorplansByName;
//Layer universes
var layersUniverses = program.mapwizeUniverseId ?
[program.mapwizeUniverseId] : [];
//Mapwize layers
var layersFromAPIByName;

function createLayer(floorplan, name, callback) {
  var layerName = 'Meraki - ' + name;
  console.log(chalk.green('\t' + layerName + '\n'));
  if (!_.has(layersFromAPIByName, layerName)) {
    var layer = {
      'name' : layerName,
      'floor' : null,
      'venueId' : program.mapwizeVenueId,
      'owner' : program.mapwizeOrganizationId,
      'universes' : layersUniverses,
      'isPublished' : false
    };

    console.log(chalk.green('\t\tDownload the Meraki floor plan
image.));
    request({url : floorplan.image_url.split('?')[0], encoding : null },
function (err, res) {
  var size = imageSize(res.body);
  var rotationAngleRad = floorplan.rotation_angle / 180 * Math.PI;
  var georeference = {
    points: [
      {
        x: size.width/2 - size.width/2*Math.cos(-rotationAngleRad)
- size.height/2*Math.sin(-rotationAngleRad),
        y: size.height/2 - size.height/2*Math.cos(-
rotationAngleRad) + size.width/2*Math.sin(-rotationAngleRad),
        longitude: floorplan.sw_lng,
        latitude: floorplan.sw_lat,
      },
      {
        x: size.width/2 + size.width/2*Math.cos(-
rotationAngleRad) + size.height/2*Math.sin(-rotationAngleRad),
        y: size.height/2 + size.height/2*Math.cos(-
rotationAngleRad) - size.width/2*Math.sin(-rotationAngleRad),
        longitude: floorplan.ne_lng,
        latitude: floorplan.ne_lat,
      }
    ]
  };
}
}
}

```

```

    ]
  }
  var TLProject = mapwize.projectPointForGeoreference([0,
size.height], georeference);
  var TRProject =
mapwize.projectPointForGeoreference([size.width, size.height], georeference);
  var BLProject = mapwize.projectPointForGeoreference([0, 0],
georeference);
  var BRProject =
mapwize.projectPointForGeoreference([size.width, 0], georeference);
  var topLeft = {latitude: TLProject[0], longitude: TLProject[1]};
  var topRight = {latitude: TRProject[0], longitude: TRProject[1]};
  var bottomLeft = {latitude: BLProject[0], longitude: BLProject[1]};
  var bottomRight = {latitude: BRProject[0], longitude:
BRProject[1]};

  layer.data = {cornersInMeraki: [{lat: topLeft.latitude, lng:
topLeft.longitude}, {lat: topRight.latitude, lng: topRight.longitude}, {lat:
bottomLeft.latitude, lng: bottomLeft.longitude}, {lat: bottomRight.latitude, lng:
bottomRight.longitude}]];

  fs.writeFile('tmp_layers/tmp.png', res.body, function (err) {
    if (err) {
      callback(err);
    }
    else {
      console.log(chalk.green('\t\tCreate the Mapwize layer.));
      MapwizeClient.createLayer(layer, function (err, layer) {
        if (err) {
          callback(err);
        }
        else {
          console.log(chalk.green('\t\tUpload the Mapwize layer
image.));

          var img = fs.createReadStream('tmp_layers/tmp.png');
          MapwizeClient.uploadLayerImage(layer._id, img,
topLeft, topRight, bottomLeft, bottomRight, function (err) {
            if (err) {
              callback(err);
            }
            else {
              fs.unlink('tmp_layers/tmp.png', callback);
            }
          });
        }
      });
    }
  });
}

```

```

        });
    }
    });
});
} else {
    console.log(chalk.red('\t\tMeraki layer < ' + layerName + ' > already
exist.));
    callback();
}
}

async.series([
    // Log into mapwize
    function (next) {
        console.log(chalk.blue('- Logging.));
        MapwizeClient.signIn(program.mapwizeUser,
program.mapwizePwd, next)
    },
    // Read JSON Meraki floor plans
    function (next) {
        console.log(chalk.blue('- Read Meraki floorplans configuration.));
        fs.readFile(program.merakiFloorPlansConfig, 'utf8', function (err,
data) {
            if(err) {
                next(err);
            }
            else {
                var json = JSON.parse(data);
                merakiFloorplansByName = _.keyBy(json, 'name');
                next();
            }
        })
    },
    //Reset Meraki layers or retrieve mapwize layers
    function (next) {
        layersFromAPIByName = {};
        MapwizeClient.getVenueLayers(program.mapwizeVenueId, function
(err, layers) {
            if (program.reset) {
                console.log(chalk.blue('- Reset all meraki layers));
                var merakiLayers = _.filter(layers, function (layer) {
                    return _.startsWith(layer.name, "Meraki")
                })
            }
            async.each(merakiLayers, function (layer, cb) {
                MapwizeClient.deleteLayer(layer._id, cb);
            });
        });
    }
]);

```

```

        }, next)
    } else {
        console.log(chalk.blue('- Retrieve all venue layers'))
        layersFromAPIByName = _.keyBy(layers, 'name');
        next();
    }
});
},
//Import floorplans to mapwize
function (next) {
    console.log(chalk.blue('- Import Meraki data into Mapwize.'));
    fs.mkdirSync('tmp_layers');
    async.eachOfSeries(merakiFloorplansByName, createLayer, function
(err) {
        if (err) {
            next(err);
        } else {
            fs.rmdir('tmp_layers', next);
        }
    });
}
], function (err) {
    if (err) {
        console.log(chalk.red('Error : ' + err));
    } else {
        console.log(chalk.green('DONE'));
    }
});
});

```

2. merakiAccessPointsToMapwize

```
#!/usr/bin/env node
```

```

var _ = require('lodash');
var async = require('async');
var fs = require('fs');
var MapwizeAPI = require('mapwize-node-api');
var chalk = require('chalk');
var program = require('commander');
var xml = require('xml2json');
var mapwize = require('./meraki-server/utils/mapwize');

```

```
program
```

```
  .version('0.1.0')
```

```
  .description('Upload Meraki access points into Mapwize')
```

```

.option('--merakiFloorPlansConfig [filepath]', 'Filepath to the Meraki
configuration for floor plans')
.option('--merakiAccessPointsConfig [filepath]', 'Filepath to the Meraki
configuration for access points')
.option('--mapwizeApiUrl [url]', 'Mapwize API URL')
.option('--mapwizeUser [user]', 'Mapwize user mail')
.option('--mapwizePwd [password]', 'Mapwize user password')
.option('--mapwizeApiKey [key]', 'Mapwize api key')
.option('--mapwizeOrganizationId [organizationId]', 'Mapwize
organizationId')
.option('--mapwizeVenueId [venueId]', 'Mapwize venueId')
.parse(process.argv)

if (!program.merakiFloorPlansConfig ||
!program.merakiAccessPointsConfig || !program.mapwizeUser ||
!program.mapwizePwd || !program.mapwizeApiKey ||
!program.mapwizeOrganizationId || !program.mapwizeVenueId) {
  console.log(chalk.red('- The arguments --merakiFloorPlansConfig, --
merakiAccessPointsConfig, --mapwizeUser, --mapwizePwd, --mapwizeApiKey,
--mapwizeOrganizationId, --mapwizeVenueId are required.'));
  return;
}

if (!program.mapwizeApiUrl) {
  program.mapwizeApiUrl = 'https://api.mapwize.io';
}

//Mapwize client
var MapwizeClient = new MapwizeAPI(program.mapwizeApiKey,
program.mapwizeOrganizationId, {serverUrl: program.mapwizeApiUrl});
//Meraki layers
var merakiLayersFromAPIByName;
//Meraki access points nodes id
var merakiAccessPointsByFloorPlanName;
//Access point from meraki config
var accessPointsByNodeIds;
//List of meraki accessPoint format mapwize
var mapwizeFormatOfMerakiAccessPoint;

function createLayerBeacons(layer, accessPoints, callback) {
  async.each(accessPoints, function (nodeIds, cb) {
    var accessPoint = accessPointsByNodeIds[nodeIds];
    if (accessPoint) {
      var xyMerakiCorners =
mapwize.getXYCorners(layer.data.cornersInMeraki);

```

```

        var xyMapwizeLayerCorner =
mapwize.getXYCorners(layer.importJob.corners);
        var accessPointCord = {lng : parseFloat(accessPoint.lng), lat :
parseFloat(accessPoint.lat)};
        var scale = mapwize.getScale(accessPointCord, xyMerakiCorners);
        var projectAccessPoint = mapwize.projectWithScale(scale,
xyMapwizeLayerCorner);

        var beacon = {
            'name' : "Meraki - " + accessPoint.name,
            'owner' : program.mapwizeOrganizationId,
            'venueId' : program.mapwizeVenueId,
            'type' : 'wifi',
            'location' : {lon : projectAccessPoint.lng, lat :
projectAccessPoint.lat},
            'floor' : layer.floor,
            'isPublished' : true,
            'properties' : { mac : accessPoint.mac },
        }
        mapwizeFormatOfMerakiAccessPoint.push(beacon);
        cb();
    }
    else {
        console.log(chalk.red("\t\tNo access point infos for beacon id < " +
nodeIds + " > name < " + accessPoint.name + " >"));
        cb();
    }
}, callback)
}

async.series([
    // Log into mapwize
    function (next) {
        console.log(chalk.blue('- Logging.'));
        MapwizeClient.signIn(program.mapwizeUser,
program.mapwizePwd, next)
    },
    //Retrieve meraki layers
    function (next) {
        console.log(chalk.blue('- Retrieve Meraki layers'));
        MapwizeClient.getVenueLayers(program.mapwizeVenueId, function
(err, layers) {
            if (err) {
                next(err);
            } else {

```



```

        merakiLayersFromAPIByName = _.keyBy(_.filter(layers,
function (layer) {
    return _.startsWith(layer.name, 'Meraki');
    }), 'name');
    next();
    }
    })
    },
    //Retrieve node_ids from floor plan meraki config file
    function (next) {
        console.log(chalk.blue('- Retrieve floor plans access points from
Meraki config file'));
        fs.readFile(program.merakiFloorPlansConfig, 'utf8', function (err,
data) {
            if (err) {
                next(err);
            } else {
                var configFileToJson = JSON.parse(data);
                merakiAccessPointsByFloorPlanName = { };
                _.forEach(configFileToJson, function (floorPlan) {
                    _.set(merakiAccessPointsByFloorPlanName, floorPlan.name,
floorPlan.node_ids.split(','));
                })
                next();
            }
        })
    },
    //Retrieve access points infos from Meraki config file
    function (next) {
        console.log(chalk.blue('- Retrieve access points from Meraki config
file'));
        fs.readFile(program.merakiAccessPointsConfig, 'utf8', function (err,
data) {
            if (err) {
                next(err);
            } else {
                var json = JSON.parse(xml.toJson(data));
                accessPointsByNodeIds = _.keyBy(json.network.access_point,
'id');
                next();
            }
        });
    },
    //Create beacons
    function (next) {

```

```

console.log(chalk.blue('- Create Access points in mapwize format '));
mapwizeFormatOfMerakiAccessPoint = [];
async.eachOf(merakiAccessPointsByFloorPlanName,      function
(accessPoints, floorPlanName, cb) {
    var layerName = "Meraki - " + floorPlanName;
    var layer = merakiLayersFromAPIByName[layerName];
    if (layer) {
        createLayerBeacons(layer, accessPoints, cb);
    } else {
        console.log(chalk.red("\tNo layer found for meraki floor plan < "
+ floorPlanName + " >"));
        cb();
    }
}, next);
},
//Sync beacons to mapwize
function (next) {
    console.log(chalk.blue("- Sync mapwize beacons"));
    var filter = function (beacon) {
        return _.startsWith(beacon.name, "Meraki - ");
    }
    MapwizeClient.syncVenueBeacons(program.mapwizeVenueId,
mapwizeFormatOfMerakiAccessPoint, {filter : filter, delete : true}, next);
}
], function (err) {
    if (err) {
        console.log(chalk.red('Error : ' + err));
    } else {
        console.log(chalk.green('DONE'));
    }
}
})

```