

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

Інститут електроенергетики

(інститут)

Факультет інформаційних технологій

(факультет)

Кафедра інформаційних технологій та комп'ютерної інженерії

(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи ступеня магістра

(бакалавра, спеціаліста, магістра)

студента Шилова Кирила Олександровича

(ПІБ)

академічної групи 123М-19-1

(шифр)

спеціальності 123 «Комп'ютерна інженерія»

(код і назва спеціальності)

за освітньо-професійною програмою «Комп'ютерна інженерія»

(офіційна назва)

на тему «Комп'ютерна система на санітарно-пропускному пункті з детальною розробкою сервера бази даних»

(назва за наказом ректора)

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	доц. Ткаченко С.М.			
розділів:				
теоретичний розділ	доц. Ткаченко С.М.			
синтез системи	доц. Ткаченко С.М.			
розроблення програмного забезпечення	ас. Бешта Л.В.			
експериментальний розділ	доц. Ткаченко С.М.			
Рецензент				
Нормоконтролер	проф. Цвіркун Л.І.			

**Дніпро
2020**

ЗАТВЕРДЖЕНО:

завідувач кафедри
інформаційних технологій
та комп'ютерної інженерії

(повна назва)

Гнатушенко В.В.

(підпис)

(прізвище, ініціали)

«__» _____ 2020 року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня магістр
(бакалавра, спеціаліста, магістра)

студенту Шилову К.О. академічної групи 123М-19-1
(прізвище та ініціали) (шифр)

спеціальності 123 «Комп'ютерна інженерія»

за освітньою-професійною програмою 123 «Комп'ютерна інженерія»
(офіційна назва)

на тему «Комп'ютерна система на санітарно-пропускному пункті з детальною розробкою сервера бази даних»,

затверджену наказом ректора НТУ «Дніпровська політехніка» від 22.10.2020 №888с

Розділ	Зміст	Термін виконання
Стан питання та постановка завдання	На основі матеріалів виробничих практик, інших науково-технічних джерел сформулювати наукове завдання, конкретизувати предмет та мету досліджень	21.09.2020
Теоретичний	Обґрунтувати теоретичну базу розв'язання наукового завдання, якому присвячено роботу	30.10.2020
Синтез системи	Розробка комп'ютерної системи	12.11.2020
Розроблення програмного забезпечення	Розробка програмного забезпечення	26.11.2020
Експериментальний розділ	Проведення і обробка результатів експериментів	06.12.2020

Завдання видано _____
(підпис керівника)

доц. Ткаченко С.М.
(прізвище, ініціали)

Дата видачі 07 вересня 2020 р.

Дата подання до екзаменаційної комісії 10.12.2020 р.

Прийнято до виконання _____
(підпис студента)

Шилов К.О.
(прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 69 с., 19 рис., 9 табл., 8 джерел, 3 додатки.

Об'єктом дослідження, є комп'ютерна система на санітарно пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем. Метою роботи, є розробка сервера бази даних, для комп'ютерної системи, що на основі даних отриманих від санітарно-пропускних пунктів надає дані необхідні для контролю роботи вентиляції.

Методом аналітичного дослідження організації праці на підприємстві харчової промисловості, було виявлено, що вірогідність одночасного перебування на території санітарної зони співробітників не є величиною з рівномірним розподілом, тобто гіпотеза про використання даних про пересування працівників між зонами підприємства для зменшення витрат на електроенергію підтвердилась.

Результатом розробки системи, є програма, що обчислює на основі даних від санітарно-пропускних пунктів питомий повітрообмін для санітарних приміщень. Галуззю застосування системи є харчові підприємства у яких велика частка праці припадає на роботу в приміщеннях із суворим санітарним регламентом. Економічна ефективність впроваджуваної системи дорівнює 18000 грн. на рік, при умові застосування системи в одному цеху, ста осіб, що перебувають на його території та SFP показчику, що дорівнює 0.55 Вт/м^3 , що означає достатньо високий ККД вентиляційної системи.

Розроблена система має достатні інтеграційні властивості для кооперування з іншими системами для забезпечення відповідності санітарним нормам.

Ключові слова: СИСТЕМА КЕРУВАННЯ, САНІТАРНИЙ ПУНКТ, БАЗА ДАНИХ, ДЕЗІНФЕКЦІЯ, ВЕТИЛЯЦІЯ, ХАРЧОВЕ ПІДПРИЄМСТВО.

ЗМІСТ

Вступ	8
1. Стан питання та постановка задачі	10
1.1 Огляд сфери та умов застосування системи	10
1.2 Огляд принципів роботи санітарно-пропускного пункту та вимоги до системи вентиляції та кондиювання повітря	11
1.2.1 Огляд застосування дезінфікуючих засобів	12
1.3 Огляд та характеристика об'єкта впровадження	14
1.4 Обґрунтування напрямку дослідження	18
1.5 Постановка задачі	19
Висновок	21
2 Теоретичний розділ	23
2.1 Аналіз обладнання санітарно-пропускного пункту	23
2.2 Класифікація хімічних дезінфікуючих засобів	26
2.3 Класифікація вентиляційних систем	28
2.4 Стратегія оптимізації процесів очистки та дезінфекції на санітарно-пропускному пункті	31
2.5 Синтез моделі керування даними	35
2.5.1 Обґрунтування методу моделювання	35
2.5.2 Вимоги методу до вирішення завдання	37
Висновок	39
3. Синтез системи керування	40
3.1 Схема функціональної структури	40
3.2 Розробка принципової схеми комп'ютерної системи на санітарно-пропускному пункті	41
3.2.1 Аналіз входів та виходів комп'ютерної системи на санітарно-пропускному пункті	41
3.2.2 Вибір елементної бази комп'ютерної системи на санітарно-пропускному пункті	42

3.2.3 Реалізація принципової схеми комп'ютерної системи на санітарно-пропускному пункті	48
Висновок	51
4. Розробка програмного забезпечення	52
4.1 Призначення та сфера використання програмного забезпечення	52
4.2 Обґрунтування технічних характеристик програми	52
4.3 Опис розробленої програми	53
4.3.1 Загальні відомості	53
4.3.2 Функціональне призначення	53
4.3.3 Опис логічної структури програми	54
4.4 Очікувані техніко-економічні показники	56
Висновок	57
5. Експериментальний розділ	58
5.1 Формулювання вимог до експерименту	58
5.2 Опис експерименту	58
5.3 Контрольний приклад для проведення випробувань	58
5.3.1 Підготовка до роботи	58
5.3.2 Дослідження працездатності та відмовостійкості програми	59
5.4 Аналіз результатів експерименту	64
Висновок	66
Висновки	67
Перелік посилань	68
Додаток А	70
Додаток Б	75
Додаток В	78

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ ТА ТЕРМІНІВ

СПП – санітарно-пропускний пункт

ISO – International Organization for Standardization, міжнародна організація, що займається випуском стандартів.

НАССР – Hazard Analysis and Critical Points, аналіз ризиків та критичні точки, концепція, що передбачає систематичну ідентифікацію, оцінку та керування небезпечними чинниками, що істотно впливають на безпечність продукції.

SFP-показчик – відображає споживану електродвигуном потужність, для створення одиничної витрати повітря у конкретній системі механічної вентиляції.

pH – міра кислотності водних розчинів.

СанПіН – санітарні правила і норми, державні підзаконні нормативно-правові акти з описаннями та вимогами безпечних та нешкідливих для людини, популяції людей та нащадків факторів місця проживання та їх оптимальних та безпечних кількісних параметрів з метою збереження здоров'я та нормальної життєдіяльності.

ККД – коефіцієнт корисної дії.

ВСТУП

Санітарний контроль є необхідною умовою роботи кожного підприємства, що займається виготовленням харчової продукції. На харчових підприємствах передбачено розділення робочих цехів на, так звані «чисті» та «технічні» зони. Пересування між цими зонами суворо регламентоване, щоб увійти у «чисту зону» підприємства кожен робітник повинен відповідним чином продезінфікувати своє взуття та руки, для того, щоб уникнути занесення на територію санітарної зони шкідливих бактерій та вірусів, та, що не менш важливо, зробити це за короткий проміжок часу. В умовах малого підприємства час на проходження з одної зони підприємства може не сильно впливати на робочий процес, але, якщо ми маємо справу з великим підприємством, черга на перехід з «технічної» зони в «чисту» суттєво знизить продуктивність праці робітника та підприємства в цілому.

Для забезпечення відповідності міжнародним стандартам якості продукції таким як ISO 9001:2015, HACCP, ISO 45001:2018 підприємство має визначити, створити та підтримувати середовище, необхідне для функціонування її процесів та досягнення відповідності вимогам до продукції та послуг, зокрема підтримання комфортних фізичних умов для працівників. Комп'ютерна система на санітарно пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем, серверна частина якої проектується в рамках цієї кваліфікаційної роботи здатна: забезпечити якісне очищення взуття та рук працівників, забезпечити відповідні параметри повітрообміну та якості повітря для необхідних епідеміологічних умов виробництва, зберігання та подальшої реалізації харчових продуктів; попереджувати негативний вплив на робітників факторів, що забруднюють повітря (вуглекислоти, пил, запахи, гази, випаровування).

Динамічність впроваджуваної системи полягає в тому, що вона реагує на зміну кількості персоналу у приміщенні та розраховує параметри для регулювання потужності вентиляторів та забезпечення відповідності вимогам охорони праці та комфортної роботи працівників.

Розробка комп'ютерної системи динамічного керування вентиляційним обладнанням на основі даних про проходження персоналу через санітарно-пропускні пункти спрямована на якісну обробку взуття та рук працівників, для відповідності санітарно-гігієнічним нормативам, зниження витрат на електроенергію, покращення якості продукції без наслідків для швидкості виробництва, зменшення ризиків захворюваності серед працівників підприємства, що в свою чергу приведе до зниження витрат на лікарняні виплати та дозволить уникнути відтоку кваліфікованих кадрів у зв'язку з захворюваннями.

Таким чином, мета цієї кваліфікаційної роботи розробка сервера бази даних, та керуючої системи, що на основі даних отриманих від санітарно-пропускних пунктів надає дані необхідні для контролю роботи вентиляції.

Об'єкт дослідження: Комп'ютерна система на санітарно пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем.

Предмет дослідження: Сервер бази даних для комп'ютерної системи санітарно-пропускного пункту база даних якого буде використана для динамічного керування вентиляційним обладнанням.

Основна задача роботи – розробка методів прийняття рішень для комп'ютерної системи знезараження та розробка сервера бази даних.

Ідея роботи – дані про пересування працівників через санітарно-пропускні пункти можуть бути використані для регуляції повітрообміну.

1 СТАН ПИТАННЯ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Огляд сфери та умов застосування системи

Харчові підприємства в Україні дуже поширені, кожне четверте підприємство на країні належить до харчової промисловості. Деякі харчові виробництва наближені до районів, де знаходяться джерела сировини, наприклад цукрові заводи або олійно-жирові підприємства. Технологічні процеси харчових виробництв досить різні, що пояснюється різноманіттям сировини, що переробляється і продукції, що виготовляється. Це вимагає використання багатьох видів устаткування і здійснення самих різноманітних процесів: подрібнення, нагрівання, сушіння, хімічної обробки, ароматизації, пресування та ін..

При всій відмінності технологічних процесів, обладнання та сировини всі харчові виробництва мають спільні риси: переробляється органічна сировина, готова продукція вживається в їжу, причому багато її видів без попередньої обробки (хлібні, кондитерські та ковбасні вироби, молочні продукти, напої та ін.). Технологічні процеси на підприємствах харчової промисловості, якість сировини і готової продукції знаходяться під постійним наглядом органів санітарного контролю, оскільки від їх стану безпосередньо залежить здоров'я населення. Пред'являються також високі вимоги до метеорологічних умов, в особливості до якості повітря, в приміщеннях харчових виробництв.

На харчових підприємствах у зв'язку з використанням багатьох видів сировини і видів її обробки мають місце практично всі види шкідливих виділень. Багато технологічних процесів супроводжуються утворенням і виділенням пилу в навколишнє середовище (хлібозаводи, олійно-жирові, тютюнові, чайні фабрики та ін.).

У повітря ряду підприємств надходять у великій кількості водяні пари (м'ясокомбінати, консервні заводи, молокозаводи та ін.). На підприємствах,

технологічний процес яких пов'язаний з бродінням, в повітря надходить діоксин вуглецю (пивоварні заводи, виноробні підприємства, виробництва дріжджів та ін.).

На багатьох харчових підприємствах застосовуються теплові процеси (нагрівання, сушка). Такі процеси поширені на хлібозаводах, цукрових заводах, кондитерських фабриках.

Для створення необхідних умов повітряного середовища, сприятливих для працівників, які забезпечують високу якість продукції, у виробничих приміщеннях підприємств харчової промисловості влаштовуються системи вентиляції, а за необхідністю підтримки строго визначених умов повітряного середовища – системи кондиціонування.

В умовах магістерської роботи не може бути розглянуто усі типи харчових підприємств, тому для створення моделі було обрано підприємство м'ясної промисловості.

Для розробки комп'ютерної системи на санітарно пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем підприємства м'ясної промисловості необхідно розглянути відповідний технологічний процес.

1.2 Огляд принципів роботи санітарно-пропускного пункту та вимоги до системи вентиляції та кондиціонування повітря

Мікроклімат характеризують наступні показники: температура, відносна вологість, швидкість руху повітря, інтенсивність теплового випромінювання. Велике значення має також чистота повітря. Зазвичай нормують мікроклімат та чистоту повітря на постійних та непостійних робочих місцях, а також у робочій

зоні. Постійне робоче місце – місце, де люди працюють біль ніж 2 години без перерв, або більше 50% робочого часу. Під робочою зоною розуміють простір, обмежений по висоті 2 метрами над рівнем підлоги, на якій знаходяться місця постійного, або непостійного перебування працівників. У таблиці 1.1 приведені оптимальні та допустимі норми температури, відносної вологості та швидкості руху повітря на постійних та непостійних робочих місцях виробничих приміщень, встановлені ДБН В.2.5-67:2013 «Опалення, вентиляція та кондиціонування»

Таблиця 1.1 – Допустимі величини температури, відносної вологості та швидкості руху повітря в робочій зоні виробничих приміщень.

Період року	Категорія робіт	Температура, °С				Відносна вологість (%) на постійних і непостійних робочих місцях	Швидкість руху повітря (м/с) на постійних і непостійних робочих місцях
		Верхня межа		Нижня межа			
		на постійних робочих місцях*	на непостійних робочих місцях*	на постійних робочих місцях*	на непостійних робочих місцях*		
Холодний період	Легка Іа	25	26	21	18	75	не більше 0,1
	Легка Іб	24	25	20	17	75	не більше 0,2
	Середньої важкості Іа	23	24	17	15	75	не більше 0,3
	Середньої важкості Іб	21	23	15	13	75	не більше 0,4
	Важка ІІІ	19	20	13	12	75	не більше 0,5
Теплий період	Легка Іа	28	30	22	20	55 при 28 °С	0,1-0,2
	Легка Іб	28	30	21	19	60 при 27 °С	0,1-0,3
	Середньої важкості Іа	27	29	18	17	65 при 26 °С	0,2-0,4
	Середньої важкості Іб	27	29	15	15	70 при 25 °С	0,2-0,5
	Важка ІІІ	26	28	15	13	75 при 24 °С	0,5-0,6

1.2.1 Огляд застосування дезінфікуючих засобів

Дезінфекції завжди повинна передувати стадія очистки поверхні. Харчові забруднення, що залишаються на недостатньо очищеній поверхні є джерелом харчування мікроорганізмів. Гарний санітарно-гігієнічний стан на харчовому підприємстві досягається за допомогою комбінованої програми ретельної очистки усіх поверхонь та обладнання з подальшою дезінфекцією. Відомо, що при ретельній очистці з поверхні видаляється до дев'яноста відсотків мікроорганізмів. На недомитій поверхні залишки забруднень не тільки захищають мікроорганізми від санітарної обробки, але й знижують ефективність дезінфікуючого засобу за рахунок ефекту розбавлення або

хімічної реакції органічної речовини та дезінфектанту. Фізична обробка взуття та рук працівників досягається апаратними властивостями санітарно-пропускного пункту.

Хімічні сполуки, призначені для використання в харчовій промисловості у якості дезінфектантів, відрізняються хімічною структурою, активністю проти різноманітних мікроорганізмів та умовами, за якими вони проявляють максимальну ефективність. У загальному випадку справедливе твердження – чим вище концентрація дезінфікуючого засобу, тим швидша та ефективніша його дія. Щоб обрати ефективний дезінфікуючий засіб потрібно експериментальним або теоретичним шляхом визначити потенційні патогенні мікроорганізми та впевнитися в тому, що обраний дезінфектант активний у відношенні до цих організмів. Оскільки хімічні дезінфектанти не мають високою проникною дією, мікроорганізми у тріщинах, подряпинах та інших нерівностях поверхні, всередині мінеральних забруднень можуть бути не повністю знищені після обробки. Для максимальної ефективності дезінфікуючих засобів поверхня повинна бути ретельно очищена.

Ефективність обробки залежить від ряду фізико-хімічних факторів:

- Час експозиції. Дослідження показали, що загибель популяції мікроорганізмів має логарифмічний характер: 90% мікроорганізмів вмирають у певний інтервал часу, 90% мікроорганізмів, що залишилися, загинуть у наступний інтервал часу, при цьому залишиться лише 1% від початкової кількості мікроорганізмів. Час експозиції залежить від ефективності дії дезінфікуючого засобу на певний вид мікроорганізмів, здатності до утворення спор та інших фізико-хімічних факторів.
- Температура. З підвищенням температури зростає швидкість росту мікроорганізмів та ефективність дії дезінфікуючих засобів. Підвищення температури призводить до зниження поверхневого натягу, в'язкості та зміні ряду інших параметрів, які сприяють загибелі мікроорганізмів.

- Концентрація. З підвищенням концентрації дезінфікуючого засобу зростає швидкість загибелі мікроорганізмів.
- Показчик рН. Активність антимікробних сполук, як правило, залежить від показника рН середовища. Наприклад, хлоро та йодовмісні речовини втрачають свою активність зі зростанням рН показника середовища.
- Жорсткість води. С підвищенням концентрації солей жорсткості води знижується біологічна активність дезінфікуючих засобів в результаті їх взаємодії з солями жорсткої води. Наприклад четверткові амонієві сполуки несумісні із солями кальцію та магнію. При жорсткості води вище 200 ppm дезінфікувати поверхні такими сполуками без додавання комплексоутворювачів, що пом'якшують воду, марно.
- Чистота поверхні та обладнання. Багато дезінфікуючих засобів – гіпохлорит, йодофори, взаємодіють із органічними сполуками, що залишаються на погано очищеній поверхні та втрачають свою біологічну активність.

1.3 Огляд та характеристика об'єкта впровадження

Комп'ютерна система на санітарно пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем підприємства м'ясної промисловості складається з кількох підсистем, що керуються сервером.

Санітарно-пропускний пункт призначений для забезпечення безконтактного дезінфікування рук та дезінфікування взуття робітника, при пересуванні між технічною зоною підприємства та промисловою зоною («чистою»). СПП має порт RJ-45 і передбачає під'єднання його до промислової мережі. Гігієнічна станція виконана у цільному сталевому (марки AISI304(1.4301)) корпусі та складається з наступних модулів:

- модуль дезінфекції рук;
- модуль умивальника;
- модуль установки сушилки чи диспенсера для паперових рушників.
- модуль аутентифікації працівника;
- модуль чистки взуття;
- модуль управління.

Технологічний лист з зображенням санітарно-пропускного пункту показаний на рисунку 1.1.

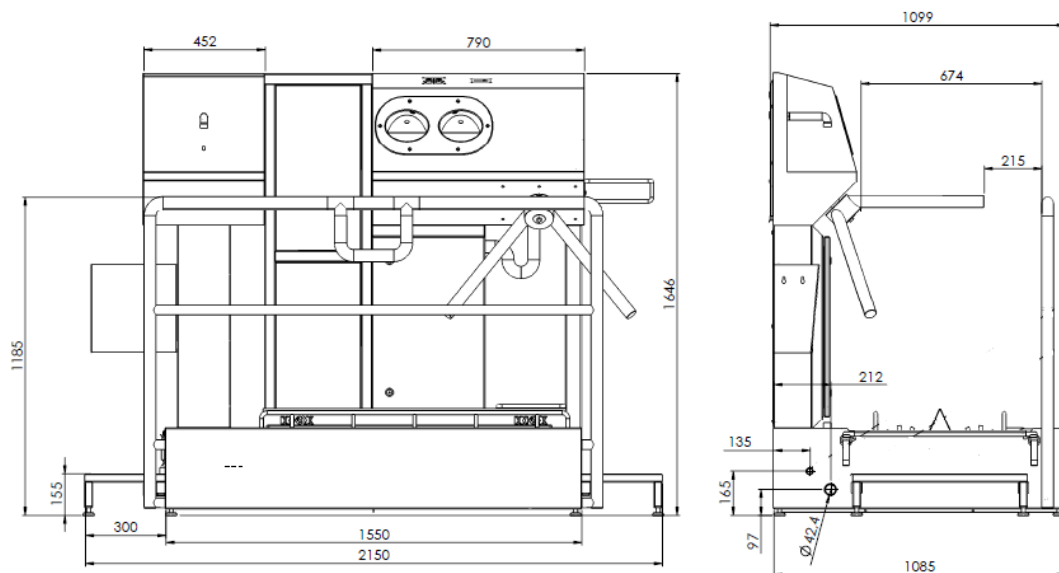


Рисунок 1.1 – Санітарно-пропускний пункт SPG 1102 D L

Конструкція передбачає обов'язкове послідовне виконання дій з обробки рук та взуття. За потреби втручання у роботу пристрою оператора (заміна витратного матеріалу, тощо), СПП сигналізує про це за допомогою звукового сигналу. Час проходження працівником санітарної обробки дорівнює 1 хвилині.

Вентиляційна система, керування якою здійснюється зокрема на основі даних сервера бази даних комп'ютерної системи санітарно-пропускного пункту, складається з відцентрових промислових вентиляторів, обладнаних

асинхронними трифазовими електродвигунами з потужністю 0.55-4.2 кВт; повітропроводів, циклонів та розподільвачів повітря, показана на рисунку 1.2.

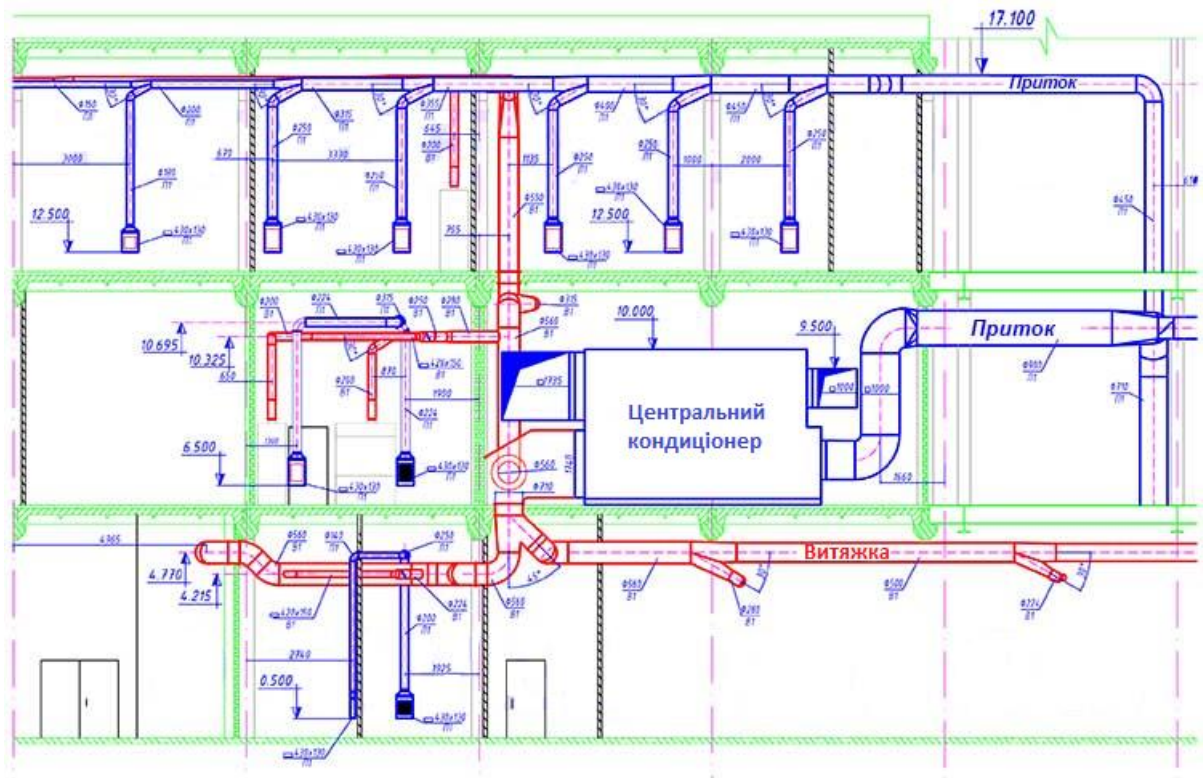


Рисунок 1.2 – Схема розташування обладнання на виробничій території

Фізичне керування відцентровими промисловими вентиляторами виконується за допомогою трифазових трансформаторних регуляторів, що мають п'ять режимів роботи. Зміна швидкості обертання трифазних вентиляторів відбувається за допомогою зміни робочої напруги на виході. Електрична схема підключення трифазного регулятора швидкості обертання вентилятора до електродвигуна представлена на рисунку 1.3

РЕ – заземлення;

М – електродвигун;

1, 2 – нерегульований вихід на 230 В;

ТК – термозапобіжник двигуна.

1.4 Обґрунтування напрямку дослідження

Керування санітарно-пропускним пунктом здійснюється за нормативним документом СанПін 42-123-5777-91. Прийняття рішень про управління вентиляційною системою зумовлюється нормативним документом ДБН В.2.5-67:2013 «Опалення, вентиляція та кондиціонування». Це діючі документи та на даний час немає причин або передумов керуватись іншими. Таким чином у рамках кваліфікаційної роботи потрібно розробити комп'ютерну систему, що збирає дані про пересування працівників між санітарною та технічною зонами та формулює відповідні параметри для керування вентиляційною системою на підприємстві м'ясопереробної промисловості.

Проектована система має збирати, обробляти та систематизувати дані про пересування працівників між санітарними та технічними зонами, вносити корективи до роботи вентиляційної системи під час робочого процесу. Стандартні вентиляційні системи мають ручне керування, тому зазвичай відбувається перевитрата електричної енергії, або, в наслідок людського фактору, може виникати дефіцит свіжого повітря у виробничому приміщенні, що є порушенням стандартів виробництва, та несе негативні наслідки як для здоров'я працівників, так і для здоров'я споживачів харчової продукції.

Таким чином, виникає необхідність створення автоматизованої комп'ютерної системи на санітарно пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує

необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем підприємства м'ясної промисловості.

1.5 Постановка задачі

Для підприємства харчової промисловості обов'язковою є відповідність вимогам до пересування між санітарними та технічними зонами та наявність ефективної вентиляційної системи, що забезпечить приплив свіжого повітря до виробничих приміщень відповідно до державних стандартів.

Для здійснення задачі управління вентиляційною системою та забезпечення необхідного повітрообміну необхідно, використовуючи чинні нормативні акти та дані про пересування працівників по підприємству, визначити критерії для режимів роботи механічної вентиляційної системи.

Мета дослідження: обґрунтувати параметри комп'ютерної системи на санітарно-пропускному пункті для визначення параметрів керування автоматизованим вентиляційним обладнанням на основі даних про пересування працівників, що надходять до бази даних від санітарно-пропускних пунктів.

Основною задачею магістерської роботи є детальна розробка сервера бази даних для комп'ютерної системи санітарно-пропускного пункту. Для рішення поставленої задачі необхідно виконати наступні пункти:

- провести аналіз обладнання, нормативних актів, а також застосовуваних методів прийняття рішень з виконання дезінфекції та повітрообміну, для виконання розробки комп'ютерної системи динамічного реагування;
- розробити автоматизовану систему збору, організації та інтерпретації даних для управління вентиляційною системою;
- розробити серверне програмне забезпечення, для збору, організації та інтерпретації даних, що необхідні для розрахунку мінімального

повітрообміну у виробничих приміщеннях, та керування режимами роботи вентиляційної системи;

- провести експериментальні дослідження методів дезінфекції на санітарно-пропускному пункті та керування вентиляційною системою та підтвердити їх ефективність.

ВИСНОВОК

Об'єкт дослідження – Комп'ютерна система на санітарно пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем підприємства м'ясної промисловості.

Предмет дослідження – сервер бази даних для комп'ютерної системи на санітарно-пропускному пункті.

Актуальність проблеми зумовлена відповідністю високій якості продукції, забезпеченням санітарно-гігієнічних умов на виробництві, високою часткою витрат на енергозабезпечення вентиляційних систем та нормативними актами з охорони праці.

Мета дослідження – обґрунтувати параметри комп'ютерної системи санітарно-пропускного пункту для коригування управління автоматизованим вентиляційним обладнанням на основі даних про пересування працівників, що надходять до бази даних від санітарно-пропускних пунктів.

Ідея роботи – дані про пересування працівників можуть бути використані для прийняття рішень з управління вентиляційним обладнанням та зберегти відповідність санітарним нормам при зниженні витрат на електроенергію.

Основна задача дослідження – розробка серверного програмного забезпечення, для збору, організації та інтерпретації даних, що надходять від санітарно-пропускних пунктів, необхідних для розрахунку мінімального повітрообміну у виробничих приміщеннях, та керування режимами роботи вентиляційної системи.

Для рішення поставленої задачі необхідно виконати наступні пункти:

- провести аналіз обладнання, нормативних актів, а також застосовуваних методів прийняття рішень з виконання дезінфекції та

повітрообміну, для виконання розробки комп'ютерної системи динамічного реагування;

- розробити автоматизовану систему збору, організації та інтерпретації даних від санітарно-пропускних пунктів для управління вентиляційною системою;
- розробити серверне програмне забезпечення, для збору, організації та інтерпретації даних, що необхідні для розрахунку мінімального повітрообміну у виробничих приміщеннях, та керування режимами роботи вентиляційної системи;
- провести експериментальні дослідження методів дезінфекції нас санітарно-пропускному пункті та керування вентиляційною системою та підтвердити їх ефективність.

2 ТЕОРЕТИЧНИЙ РОЗДІЛ

2.1 Аналіз обладнання санітарно-пропускного пункту

Структура комплексу технічних засобів санітарно-пропускного пункту відображено на рисунку 2.1

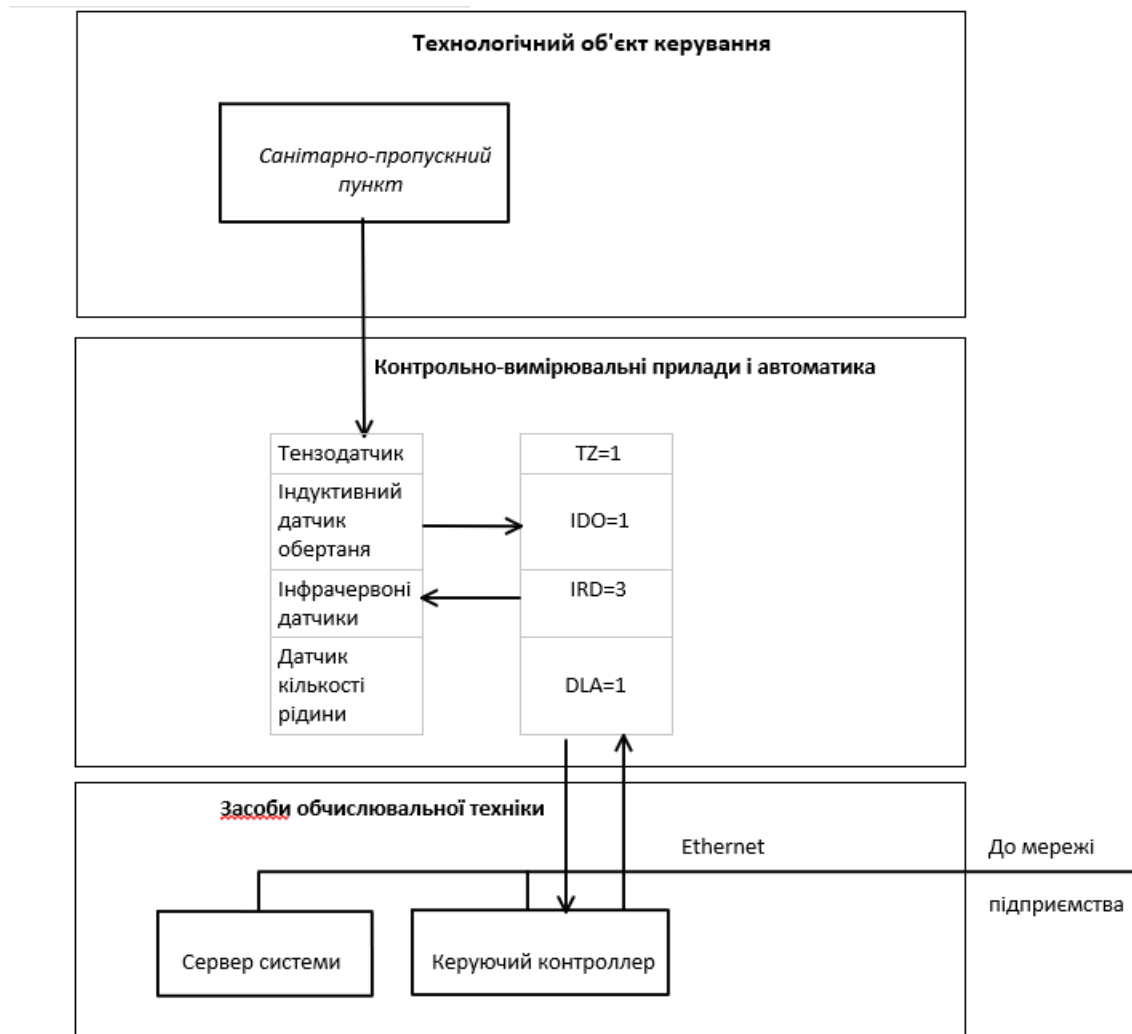


Рисунок 2.1 – Структура комплексу технічних засобів комп'ютерної системи на санітарно-пропускному пункті

Інфрачервоні датчики руху. Конструкція PIR датчика руху не дуже складна – він складається з піроелектричного елемента, який відрізняється високою чутливістю (деталь циліндричної форми, в центрі якої знаходиться

кристал) до наявності у зоні дії певного рівня інфрачервоного випромінювання. Зверху PIR-датчика встановлюється півсфера, розділена на декілька ділянок (лінз), кожна з яких забезпечує фокусування випромінювання теплової енергії на різноманітні сегменти датчика руху. Частіше за все в якості лінзи застосовують лінзу Френеля, яка за рахунок концентрації теплового випромінювання дозволяє розширити діапазон чутливості інфрачервоного датчика руху Arduino. Датчик руху Arduino зображений на рисунку 2.2.



Рисунок 2.2 – Інфрачервоний датчик руху

PIR-sensor конструктивно розділений на дві половини. Це обумовлено тим, що для влаштування сигналізації важлива саме наявність руху в зоні чутливості, а не сам рівень випромінювання. Тому частини влаштовані таким способом, що при уловлюванні однією більшого рівня випромінювання, на вихід буде подаватися сигнал зі значенням high або low.

Основними технічними характеристиками датчика руху Arduino є:

- Зона виявлення рухомих об'єктів становить від 0 до 7 метрів;
- Діапазон кута стеження – 110°;
- Напруга живлення – 4.5-6 В;
- Робочий струм – до 0.05 мА;
- Температурний режим – від -20° до +50°С;
- Регульований час затримки від 0.3 до 18 с.

Модуль, на якому встановлено інфрачервоний датчик руху включає додаткове електричне обладнання з запобіжниками, резисторами й конденсаторами.

Індуктивний датчик повороту турнікета (Абсолютний багатооборотний енкодер). Перевагою абсолютного енкодера є те, що він більш стійкий до перешкод. У випадках, коли сигнал с диску не може бути повноцінно зчитаний енкодером, наприклад, якщо вал обертається занадто швидко, точний кут оберту буде зчитано, коли швидкість оберту зменшиться. Але навіть при швидкому обертанні валу можлива реєстрація положення, хоча й з меншою точністю. Для цього приймається до уваги тільки інформація від більш старших розрядів, тобто змінена (за рахунок великої швидкості обертання) інформація яка надходить від молодших розрядів просто ігнорується. Абсолютний енкодер стійкий до вібрацій та іншого типу перешкод, тому, що похибка в результаті підрахунку “помилкових” імпульсів, які виникають в результаті, наприклад, вібрації, виключена. Індуктивний датчик обертання турнікету зображений на рисунку 2.3.



Рисунок 2.3 – Індуктивний датчик обертання турнікету

Технічні характеристики датчика:

- Вал/Полий вал \varnothing : \varnothing 6/ 1/4" / 8 / 10 мм;
- Діаметр корпусу: \varnothing 36x42 мм;
- Макс. швидкість: 12.000 об/хв.;
- Темп. навк. серед: -30°C ... +85°C;

- Ступінь захисту: IP 67;
- Тип з'єднання: Кабель;
- Роздільна здатність: 16 Біт;
- Інтерфейс: CANopen;
- Напруга жив. DC: 10...30 V;
- Опціонально: EX, дозволено використовувати в небезпечних зонах 2/22.

Датчик рівня рідини BL13-TDT зображений на рисунку 2.4.



Рисунок 2.4 – Датчик рівня рідини

Датчики рівня рідини серії BL визначають наявність чи відсутність рідини у прозорій трубці діаметром 6–13 мм с товщиною стінки 1 мм. Вони працюють у двох режимах (на світло чи затемнення), що можна вибрати за допомогою кнопочного перемикача, і світлодіодні індикатори (зелений та червоний) для зручного контролю поточного робочого стану. Компактні розміри виробів серії BL дозволяють економити монтажний простір. Температурний режим +10...+55 °С - використання, зберігання: -25...+65 °С. Тип виходу NPN-вихід с відкритим колектором. Має захист від переполюсовки, короткого замикання та надструму на виході. Ступень захисту IP64 (стандарт МЭК).

2.2 Класифікація хімічних дезінфікуючих засобів

Дезінфікуючі засоби класифікують за їх дією на різні форми мікроорганізмів: бактерициди знищують вегетативні мікроорганізми, спороциди знищують спори, фунгіциди знищують гриби, вірусициди знищують

віруси. Хімічні антисептики використовуються для дезінфекції шкіри. Бактеріостатичні речовини перешкоджають розмноженню бактерій, фактично не знищуючи їх. Звичайно хімічні дезінфікуючі засоби класифікують по типу складової біологічно-активної речовини. Виділяють вісім основних видів дезінфектантів: галогеновмісні, четверткові амонієві сполуки, перекис водню, надуксусна кислота, спирти, альдегіди, бісфеноли та бігуанідіни. Порівняльний аналіз основних видів дезінфектантів наведено у таблиці 2.1

Таблиця 2.1 – Основні хімічні дезінфікуючі речовини

Дезінфікуючі агенти	Дозування	Активність у відношенні бактерій	Активність у відношенні спор	Коментарі
галогеновмісні	50-250 мг/л	> 10 мг/л	> 50 мг/л	Хлоровмісні сполуки дешевші ніж йодофори, але мають корозійні властивості
четверткові амонієві сполуки	150-250 мг/л	> 100 мг/л	–	Мають пролонговану дію, нейтральні, неагресивні
перекис водню	3-90%	>6%	10-30%	Більш ефективна у сполученні з надуксусною кислотою

Продовження таблиці 2.1

надуксусна кислота	30-250 ppm	30 ppm	> 100 ppm	Широкий спектр активності, присутність органічних речовин практично не зменшує активності
спирти	20-70%	>22%	60-70%	Мають промислове застосування
альдегіди	0.8-0.16 мг/л	< 10 мг/л	20 мг/л	Мають обмежене застосування у харчовій промисловості
бісфеноли	2-20 мг/л	> 10 мг/л	–	
бігуанідини	>150 мг/л	1-60 мг/л	–	Використовують у рецептурах антисептиків для шкіри

2.3 Класифікація вентиляційних систем

Системи вентиляції забезпечують підтримання у приміщеннях різноманітного призначення допустимих параметрів повітряного середовища. Оптимальні параметри можуть створюватись у приміщеннях із системами кондиціонування.

При всій різноманітності систем вентиляції, зумовленій призначенням приміщень, характером технологічного процесу, видом шкідливих виділень,

задачами, що ставляться перед вентиляційними системами та ін., їх можна класифікувати за наступними характерними властивостями: за призначенням (витяжні та припливні), сфері дії (місцеві та загальнообмінні), засобу створення тиску для пересування повітря (природним та механічним), конструктивним властивостями (канальні та безканальні).

Витяжні системи призначені для видалення з приміщень забрудненого повітря. Припливні системи подають у вентилязовані приміщення чисте повітря замість видаленого. Це повітря за необхідності піддається спеціальній обробці (очистці, нагріванню, зволоженню, тощо). В загальному випадку у приміщенні передбачаються як припливні так і витяжні системи.

Місцеві системи вентиляції обслуговують обмежені ділянки приміщень. Отже, місцеві витяжні системи, які зазвичай називають локалізуючи ми, видаляють забруднене повітря від технологічного обладнання або інших джерел виділення шкідливих речовин. Системи, що локалізують пиловиділення називають аспірацією. Місцеві припливні системи подають повітря в окремі точки приміщення, наприклад на робочі місця, на яких потрібно створити певні метеорологічні умови. Місцеві витяжні системи, як правило, досить ефективні, тому що дозволяють видалити шкідливі речовини безпосередньо від місця їх утворення або виділення, не даючи їм поширитися в приміщенні. Завдяки значній концентрації шкідливих речовин (парів, газів, пилу) в повітрі зазвичай вдається при невеликому обсязі повітря що вилучається досягти хорошого санітарно-гігієнічного ефекту.

Однак місцеві системи не можуть вирішити всіх завдань, що стоять перед вентиляцією: не всі шкідливі речовини можуть бути локалізовані цими системами, наприклад шкідливі виділення, що знаходяться на великій площі або об'ємі; подача повітря в окремі точки приміщення не може забезпечити необхідні умови повітряного середовища, якщо робота проводиться на всій площі приміщення або її характер пов'язаний із переміщенням.

Загальнообмінні витяжні системи відносно рівномірно видаляють повітря зі всього приміщення. Загальнообмінні припливні системи подають повітря і розподіляють його по всьому вентиляваному приміщенню.

Переміщення повітря в системах вентиляції відбувається під дією природного тиску, що виникає внаслідок різниці температур зовнішнього повітря і повітря в приміщенні, в результаті впливу на будівлю вітру, а також при суспільній дії цих факторів (природна вентиляція), або під дією тиску, що створює вентилятор (механічна вентиляція). Системи природної вентиляції застосовують у вигляді аерації і каналних гравітаційних систем. Системи природної вентиляції прості, не вимагають складного дорогого устаткування, витрати електроенергії. Однак залежність дії цих систем і їх ефективності від змінних факторів (температури повітря, напрямку і швидкості вітру), а також слабкий тиск не дозволяють вирішити з їх допомогою всі складні і різноманітні завдання у сфері вентиляції приміщень.

У механічних системах використовується складне і дороге устаткування та прилади (вентилятори, електродвигуни, повітрянагрівачі, пиловловлювачі та інші). Витрати електроенергії на їх функціонування складають вагомую частку в її загальній витраті – до 10-15 % від загальної витрати підприємством. Ці системи можуть подавати і видаляти повітря за заданих точок в необхідній кількості незалежно від мінливих обставин навколишнього середовища. За необхідності повітря піддають різним видам обробки (очищенню, нагріванню, зволоженню і т.д.), що практично неможливо у системах природної вентиляції.

Системи вентиляції мають розгалужену систему повітропроводів для переміщення повітря (каналні системи), або канали (повітропроводи) можуть бути відсутні, наприклад при аерації, або установці вентиляторів у стіні, перекритті, тощо.

Для сучасних припливно-витяжних установок витрати електроенергії на виробництво 1 м^3 повітря (показник SFP) дорівнюють $0.55\text{ Вт/м}^3/\text{год}$.

Кількість вентиляційного повітря визначається для кожного приміщення окремо з урахуванням наявності шкідливих речовин. Якщо характер та кількість шкідливих речовин не піддаються обліку, повітрообмін визначають за мінімальною питомою витратою повітря на людину.

Мінімальну питому витрату вентиляційного повітря на людину для виробничих приміщень слід приймати згідно таблиці 2.2 з додатку X ДБН В.2.5-67:2013[]

Таблиця 2.2 – Мінімальна питома витрата вентиляційного повітря для виробничих приміщень

Приміщення (ділянка, зона)	Приміщення	
	З природним провітрюванням	Без природного провітрювання
Виробничі	30 м ³ /год	60 м ³ /год

2.4 Стратегія оптимізації процесів очистки та дезінфекції на санітарно-пропускному пункті

Стійкість патогенних мікроорганізмів до дії миючих та дезінфікуючих речовин по ще не стала глобальною проблемою харчової промисловості. Однак робітники харчової промисловості повинні усвідомлювати можливість появи стійких штамів патогенних мікроорганізмів у випадку неправомірного використання миючих та дезінфікуючих засобів. Навіть короточасний вплив дезінфікуючими речовинами на *Listeria Monocytogenes* у концентраціях недостатніх для загибелі патогенних мікроорганізмів викликає їх мутацію, при цьому можуть бути сформовані патогенні мікроорганізми, стійкі до дії дезінфікуючих речовин. При використанні миючих та дезінфікуючих засобів слід приймати до уваги наступні фактори:

- Вибір дезінфектанту, активного проти даного виду патогенних мікроорганізмів;
- Дотримання умов застосування, рекомендованих виробником;
- Наявність речовин, що здатні деактивувати миючі та дезінфікуючі засоби;
- Моніторинг чистоти поверхонь та мікробного фону перед та після використання відповідних речовин.

Важливим є вірний вибір дезінфікуючого агента, що має необхідний спектр активності проти конкретного типу мікроорганізмів. Наприклад, не має сенсу використовувати дезінфікуючий агент на основі спиртів проти спороутворюючих бактерій внаслідок низької ефективності та можливості мутації деяких мікроорганізмів. Використання дезінфікуючих агентів, що виділяють активний хлор, буде обмежено їх властивістю до корозії металів та їх здатністю руйнувати матеріали на полімерній основі.

Виконання умов застосування, що забезпечують максимальний ефект зниження числа мікроорганізмів, є важливою обставиною як з точки зору безпеки харчового виробництва, так і з точки зору зменшення фінансових витрат на прибирання та дезінфекцію. До основних факторів, що впливають на процеси мийки та дезінфекції, відносять концентрацію миючих або дезінфікуючих речовин, механічну роботу, час та температуру. При застосуванні дезінфікуючих засобів необхідно дотримуватись рекомендацій виробника. Використання більш концентрованих розчинів, ніж це необхідно, може привести до виникнення нерозчинних сполук та активації корозійних процесів. Обробка поверхонь та обладнання за температур більш високих, ніж рекомендовано в інструкції виробника може привести до хімічного розпаду активної речовини, випадінню солей жорсткості води, полімеризації протеїнів та жирів, що негативно відобразиться на якості дезінфекції. Ще одним важливим чинником у процесі дезінфекції є час контакту дезінфікуючої речовини, чим довше контакт, тим вище число інактивованих мікроорганізмів.

Існує багато математичних моделей, що описують процес інактивації мікроорганізмів. При використанні цих моделей виникають труднощі, оскільки вони містять в собі параметри, які важко визначити експериментальним шляхом. Одна з моделей, що використовується частіше за інші, є лінійно-логарифмічна модель Чіка-Ватсона (2.1)

$$\text{Log}(N_1/N_0) = -k \cdot C \cdot n \cdot t \quad (2.1)$$

N_1 – кількість мікроорганізмів, що вижило;

N_0 – початкова кількість мікроорганізмів;

k – константа швидкості дезінфекції;

C – концентрація дезінфектанту;

n – коефіцієнт розбавлення;

t – тривалість контакту.

Коефіцієнт розбавлення n залежить від типу дезінфікуючого агента. Наприклад, для четверткових амонієвих солей $n=1$. Це означає, що при зниженні концентрації у два рази, потрібно у два рази збільшити тривалість експозиції. Для етанолу $n=10$, тому при зменшені концентрації у два рази, ефективність обробки зменшиться у $2^{10} = 1024$ рази.

Механічна робота вносить істотний вклад у якість дезінфекційної обробки. Чим більше докладено механічних зусиль, тим меншою буде кількість мікроорганізмів після очистки. Кінцевий результат залежить від правильності вибору дезінфектанту та технологій мийки та дезінфекції. При виборі миючих та дезінфікуючих засобів та визначенні черговості виконання технологічних операцій слід обирати таке поєднання миючих та дезінфікуючих засобів, щоб не відбувалося інактивації дезінфікуючої речовини.

Характеристики стандартних дезінфікуючих речовин наведено у таблиці 2.3.

Таблиця 2.3 - Характеристики стандартних дезінфікуючих речовин

Характеристики	Йодофори	Хлоровмісні сполуки	Кислотовмісні сполуки	Четверткові амонієві солі
Антимікробна активність	Веgetативні клітини	Хороша	Хороша	Хороша, є певна селективність
Вплив на дріжджі	Хороша	Хороша	Хороша	Хороша
Вплив на цвілеві гриби	Хороша	Хороша	Хороша	Хороша
Стабільність при зберіганні	Залежить від температури	Низька	Дуже висока	Дуже висока
Стабільність при використанні	Залежить від температури	Залежить від температури	Дуже висока	Дуже висока
Швидкість дії	Швидка	Швидка	Швидка	Швидка
Проникна здатність	Хороша	Слабка	Хороша	Дуже висока
Плівкоутворення	Немає або слабке	–	–	+
Вплив органічних речовин	Середній	Сильний	Незначний	Незначний

Виходячи з даних, що наведені у таблиці 2.3 має сенс ввести вимогу до використання для дезінфекції рук та підшов взуття працівників розчинами на

основі четверткових амонієвих солей, що зумовлено їх високими показниками як при експлуатації, так і при зберіганні.

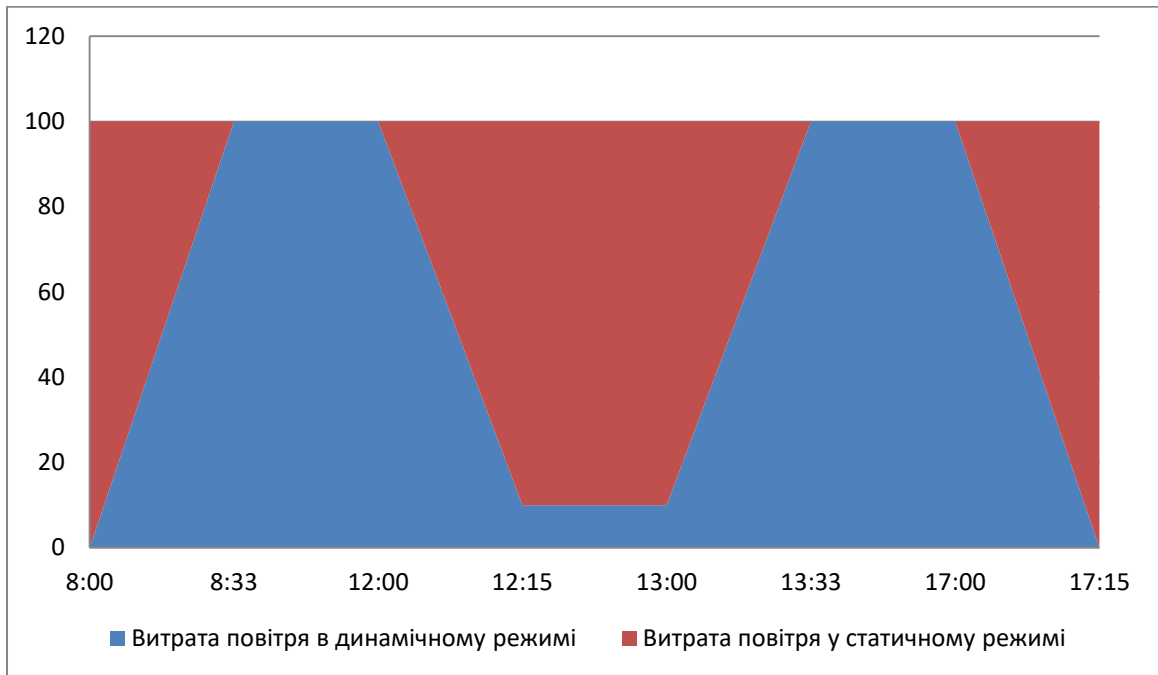
2.5 Синтез моделі керування даними

2.5.1 Обґрунтування методу моделювання

Аналіз критеріїв, що викладені у попередніх пунктах дозволив виділити наступні залежності:

- 1) Ймовірність проходження працівниками санітарно-пропускних пунктів нелінійно розподілена за часом;
- 2) Інтенсивність обертання пропелерів витяжних систем прямо пропорційна кількості людей у приміщенні;
- 3) Мінімальна питома витрата вентиляційного повітря для виробничих приміщень залежить від розподілу часу проходження працівниками санітарно-пропускних пунктів та кількості робітників у санітарному приміщенні;
- 4) Пропускна здатність санітарно-пропускного пункту дорівнює 60 чоловік у годину.

На підставі попередніх пунктів можна скласти графік залежності мінімальної питомої витрати вентиляційного повітря та інтенсивності роботи витяжної системи від часу та кількості людей, що знаходяться у приміщенні. (2.1)



Графік 2.1 – Графік залежності мінімальної питомої витрати вентиляційного повітря та інтенсивності роботи витяжної системи від часу та кількості людей, що знаходяться у приміщенні.

На графіку 2.1 червоним кольором проілюстрована мінімальна питома витрата повітря на задоволення санітарним нормам упродовж робочого дня для 100 працівників, що знаходяться у приміщенні, а синім, проілюстрована мінімальна питома витрата повітря при динамічному відстеженні кількості працівників у приміщенні за допомогою проектованої комп'ютерної системи на санітарно-пропускному пункті.

Графік побудований з розрахунком: робочих часів з 8:00 по 12:00 для першої зміни працівників; з 12:00 по 13:00 обідня перерва, що поєднана з плановим прибиранням та дезінфекцією на території цеху; з 13:00 по 17:00 друга робоча зміна; та пропускної здатності трьох станцій гігієни 60 чол/год на вхід та 133 чол/год на вихід.

Провівши аналіз графіку, можна вдатися до аналітичного методу моделювання системи, що характеризує систему з боку витрати електроенергії (забезпечення мінімальної питомої витрати повітря).

2.5.2 Вимоги методу до вирішення завдання

Аналітичне моделювання передбачає використання математичної моделі реального об'єкту у формі алгебраїчних, диференціальних, інтегральних та інших рівнянь, що пов'язують вхідні змінні з вихідними, доповнені системою обмежень. При цьому передбачається наявність однозначної обчислюваної процедури одержання точного рішення рівнянь. Виходячи з графіку та попередньо розглянутих підрозділів можна вивести наступну систему рівнянь (2.2) для розрахунку орієнтовної добової ефективності системи:

$$\left\{ \begin{array}{l} W = 3w \times \Delta t \times SFP, 8:00 < t \leq 8:33 \\ W = N \times w \times SFP, 8:33 < t \leq 12:00 \\ W = (N_{max} - 2.21\Delta t)w \times SFP, 12:00 < t \leq 12:15 \\ W = N_{приб} \times w \times SFP, 12:15 < t \leq 13:00 \\ W = 3w \times \Delta t \times SFP, 13:00 < t \leq 13:33 \\ W = N \times w \times SFP, 13:33 < t \leq 17:00 \\ W = (N_{max} - 2.21\Delta t)w \times SFP, 17:00 < t \leq 17:15 \\ W_{max} > N_{max} \times SFP \times w \end{array} \right. \quad (2.2)$$

- W – витрата електроенергії (Вт);
- w – мінімальний повітрообмін на людину ($\text{м}^3/\text{год}$);
- N – кількість людей у приміщенні;
- N_{max} – максимальна розрахована кількість працівників;
- t – час;
- SFP – витрата електроенергії на виробництво 1м^3 повітря за годину ($\text{Вт}/\text{м}^3/\text{год}$);
- W_{max} – максимальна потужність вентиляційної системи.

Аналітична модель, складена на основі розгляду кількості працівників у приміщенні як випадкової величини, дозволяє провести попередній аналіз витрати електроенергії при динамічному керуванні системи вентиляції за допомогою комп'ютерної системи на санітарно-пропускному пункті.

Також, шляхом аналізу характеристик стандартних дезінфікуючих засобів, було сформовано вимогу використання на санітарно-пропускних пунктах розчинів на основі четверткових амонієвих солей, зумовлену їх високою проникною здатністю.

ВИСНОВОК

В результаті аналізу обладнання, дезінфікуючих засобів, вимог до вентиляції та санітарної обробки приміщень на харчовому підприємстві м'ясообробної промисловості було обґрунтовано вирішення задачі шляхом аналітичного моделювання комп'ютерної системи на санітарно-пропускному пункті, на основі даних якого виконується динамічне керування вентиляційним обладнанням, що підтримує необхідну інтенсивність повітрообміну на основі даних про пересування персоналу між технічними зонами та зонами із підвищеним санітарним контролем.

Згідно методу аналітичного моделювання було використано математичну модель реального об'єкту у формі алгебраїчних рівнянь, що пов'язують вихідні змінні з вхідними та доповнено системою обмежень.

Сформульоване наукове положення для прийняття рішень по керуванню, на основі даних про пересування працівників між санітарною та технічною зонами, вентиляційним обладнанням, необхідно та достатньо використовувати дані, що надходять до серверу від санітарно-пропускних пунктів, та обчислені залежності витрат електроенергії на забезпечення мінімального питомого повітрообміну на працівника.

3 СИНТЕЗ СИСТЕМИ КЕРУВАННЯ

3.1 Схема функціональної структури

Комп'ютерна система на санітарно-пропускному пункті повинна виконувати наступні функції:

- Контроль доступу працівників до санітарних зон підприємства;
- Дезінфекція та очистка рук та підошов взуття працівників;
- Збір, обробку та занесення до бази даних інформації, про проходження робітниками підприємства санітарно-пропускних пунктів;
- Розрахунок мінімального питомого повітрообміну у приміщенні на основі даних про пересування працівників;
- Розрахунок потужності вентиляційної системи відповідно до інформації про кількість працівників, що знаходяться у приміщенні.

Виходячи з переліку приведених функцій схема функціональної структури буде виглядати як показано на рисунку 3.1.

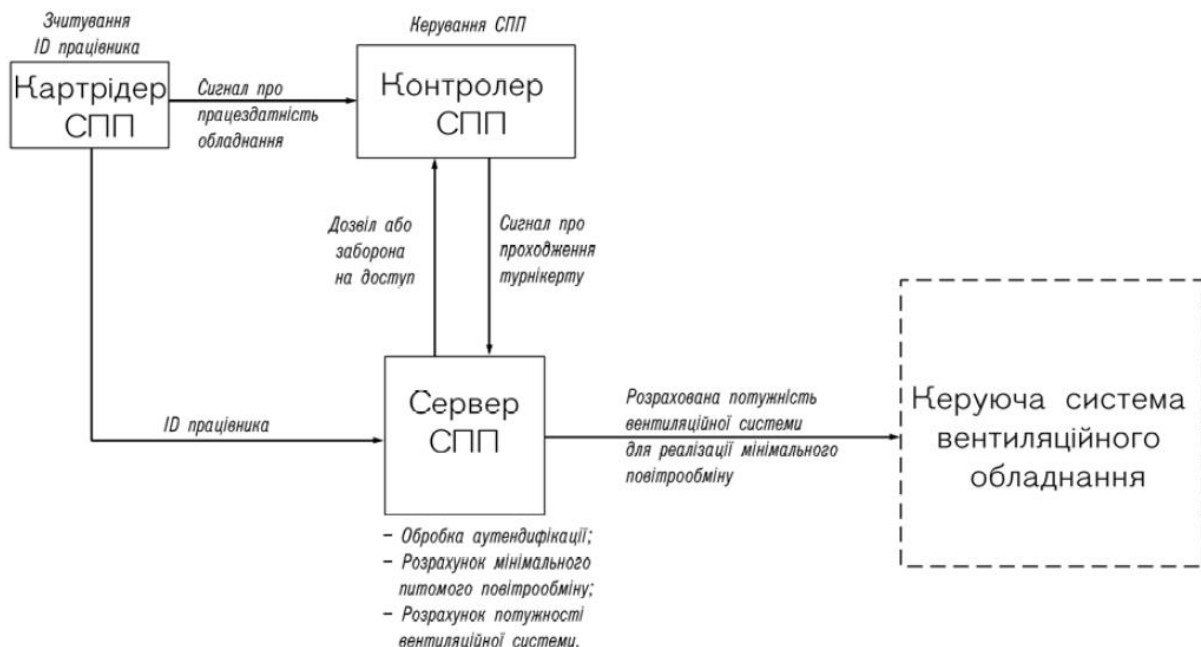


Рисунок 3.1 – Схема функціональної структури комп'ютерної системи на санітарно-пропускному пункті

Приведена схема функціональної структури комп'ютерної системи на санітарно-пропускному пункті дозволяє розробити принципову схему системи.

3.2 Розробка принципової схеми комп'ютерної системи на санітарно-пропускному пункті

3.2.1 Аналіз входів та виходів комп'ютерної системи на санітарно-пропускному пункті

Для визначення входів та виходів комп'ютерної системи на санітарно-пропускному пункті необхідно провести аналіз та класифікацію входів та виходів датчиків та виконавчих механізмів технологічного обладнання.

В результаті аналізу, отримана класифікація входів та виходів, що наведена таблиці 3.1.

Таблиця 3.1 – Перелік вхідних та вихідних сигналів КС на СПП

п.п.	Найменування сигналу	Умовне позначення сигналу (Ідентифікатор)	Напрямок вхід/вихід	Функція	Джерело/Отримувач	Форма представлення (розрядність, діапазон)		Період вводу/виводу
						Зовнішня	Внутрішня	
1	Сервер(база даних)-плк	SQ1 (1-1)	вхід	контр.	картридер/плк	0/1, 24В	1 біт	прогр.
2	Датчик умивальника	SQ2 (2-1)	вхід	контр.	датчик/плк	0/1, 24В	1 біт	постійн
3	Датчики діз.камери	SQ3 (3-1)	вхід	контр.	датчик/плк	0/1, 24В	1 біт	постійн

Продовження таблиці 3.1

4	Датчик повороту турнікету	SQ4 (4-1)	вхід	контр.	датчик/плк	0/1, 24В	1 біт	постійн
5	Датчик кількості рідини	SQ5 (5-1)	вхід	контр.	датчик/плк	0/1, 24В	1 біт	постійн
6	Картрідер-база даних	NS10 (10-1)	вихід	керув.	картрідер/сервер	24В	25 байт	прогр.
7	Таймер на перевищення часу відгуку	NS11 (11-1)	вихід	керув.	плк/реле	24В	1 біт	прогр.
8	Таймер на завершення операції	NS12 (12-1)	вихід	керув.	плк/реле	24В	1 біт	прогр.
9	Подача води	NS13 (13-1)	вихід	керув.	плк/реле	24В	1 біт	прогр.
10	Робота щіток	NS14 (14-1)	вихід	керув.	плк/реле	24В	1 біт	прогр.
11	Дозвілна прохід	NS15 (15-1)	вихід	керув.	плк/реле	24В	1 біт	прогр.
12	Кнопка «ПУСК/СТОП»	HS1	вхід	контр.	кноп/плк	0/1, 24В	1 біт	постійн

Технічна специфікація комп'ютерної системи на санітарно-пропускному пункті може бути використана для вибору елементної бази системи керування.

Перелік вхідних та вихідних сигналів комп'ютерної системи на санітарно-пропускному пункті може бути використаний для математичного опису програмного забезпечення системи.

3.2.2 Вибір елементної бази комп'ютерної системи на санітарно-пропускному пункті

Для комп'ютерної системи санітарно-пропускного пункту вибрано контролер VIPA System 200V. Контролери VIPA System 200V логічні контролери з можливістю розширення, призначені для використання в якості віддалених інтелектуальних вузлів у розподілених мережах технологічного рівня, а також для автономної роботи пристроїв керування невеликими установками, в цьому випадку роботою приводів санітарної станції.

System 200V — найбільш розвинене сімейство контролерів VIPA для розв'язання завдань централізованої й розподіленої систем автоматизації. Вони використовуються в системах промислової автоматизації з підвищеними вимогами до надійності встаткування й до тимчасових параметрів контурів керування. Серія System 200V побудована за модульним принципом, що дозволяє оптимально підбирати конфігурацію для розв'язання того або іншого конкретного завдання й легко модифікувати систему при її розширенні або зміні вимог до неї.

Всі модулі вводу-виводу й інтерфейсні модулі універсальні, що дозволяє сполучати їх з будь-яким центральним процесором (CPU) даної серії. При цьому є можливість вибору процесорного модуля з оптимальною продуктивністю для розв'язку конкретного завдання.

Контролери VIPA серії System 200V мають невеликий час відгуку й підходять для керування виробництвами періодичного, безперервного й безперервно-періодичного типу.

На підставі переліку вхідних і вихідних сигналів та їх параметрів обрано процесорний модуль VIPA 214-1BT10, рисунок 3.2.



Рисунок 3.2 – Процесорний модуль VIPA 214-1BT10

Технічні характеристики процесорного модуля VIPA 214NET (214-1BT10):

- модульна конструкція: дискретні I/O – до 32, аналогові I/O – до 16, інтерфейсні, комунікаційні, живлення.
- обсяг пам'яті постійна/оперативна: 48/80 кБайт;
- тип пам'яті: RAM+FLASH;
- розширення пам'яті: MMC до 64Мб;
- час виконання операції: з бітами - 0,18 мкс.
- інтерфейс програмування: MP2I (MPI+PPI);
- тип інтерфейсу (електричний): RS485, RJ45;
- протоколи, що підтримуються: MPI, TCP/IP;
- підтримка мереж: Ethernet-CP 243, TCP/IP, S7-протокол и RFC1006.

Для підключення датчиків до контролера вибрано модуль дискретних входів VIPA 221-1BH30 (Рисунок 3.3);

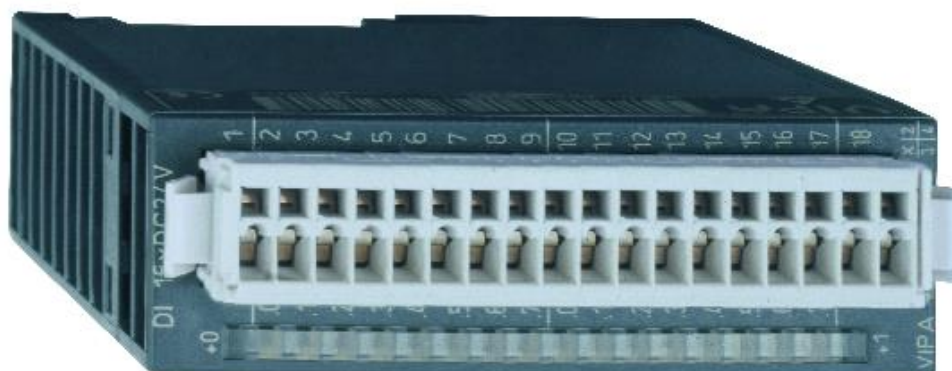


Рисунок 3.3 – Модуль дискретних входів VIPA 221-1BH30

Характеристики модуля дискретних входів VIPA 221-1BH30:

- Кількість входів – 16;
- Тип вхідного сигналу 24В, постійного струму.
- Температура оточуючого середовища 0 °С .. 60 °С
- Габаритні розміри, В x Ш x Г, 25.4мм x 76мм x 88мм

Для підключення виконавчих пристроїв до контролера вибрано модуль дискретних виходів VIPA 222-1BH20 (Рисунок 3.4).

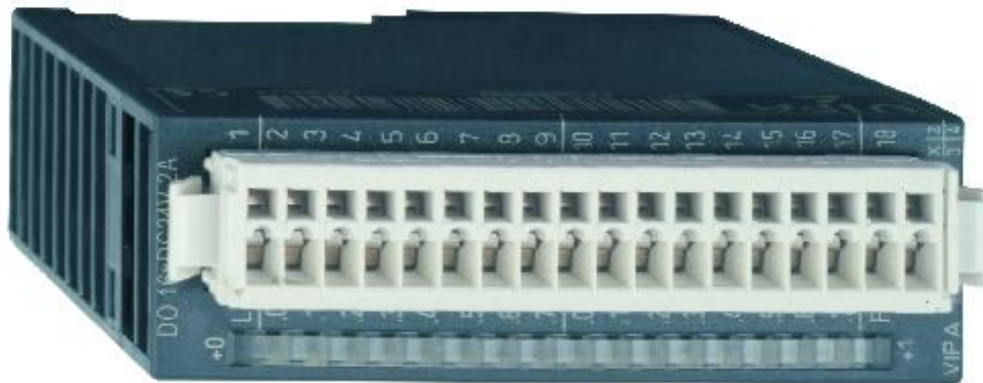


Рисунок 3.4 – Модуль дискретних виходів VIPA 222-1BH20

Характеристики модуля дискретних виходів VIPA 222-1BH20:

- кількість виходів – 16;
- Тип вихідного сигналу 24В, постійного струму, 2А, сумарний струм до 10А.
- Температура оточуючого середовища 0 °С .. 60 °С
- Габаритні розміри, В x Ш x Г, 25.4мм x 76мм x 88мм

Для забезпечення роботи контролера вибрано джерело живлення VIPA 207-1BA00 (Рисунок 3.5).

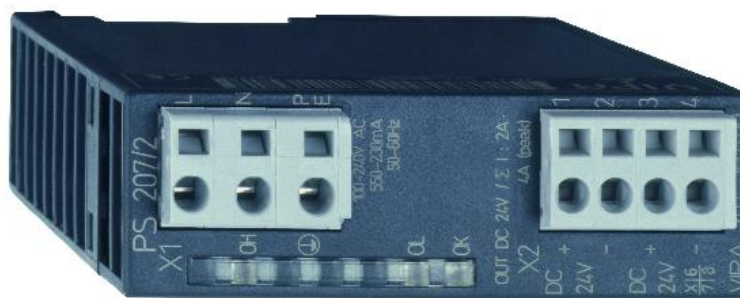


Рисунок 3.5 – Джерело живлення VIPA 207-1BA00

Технічні характеристики джерела живлення VIPA 207-1BA00:

- Вхідна напруга 100..240 В;
- Вихідна напруга 24 В;
- Вихідний струм, max 2А;
- Захист виходу від перенапруги та короткого замикання.

Підключення виконавчих пристроїв до виходів контролера здійснюється за допомогою електромагнітного реле.

Результати вибору пристроїв керування та їх модулів приведено у таблиці 3.2.

Таблиця 3.2 – Пристрої керування та їх модулі

№	Назва модулю	Пристрій
1	VIPA 214NET (214-1BT10)	Процесорний модуль
2	VIPA 221-1BH30	Модуль дискретних входів
3	VIPA 222-1BH20	Модуль дискретних виходів
4	VIPA 207-1BA00	Джерело живлення

Основними функціями сервера нижнього рівня є збір інформації про переміщення персоналу, надання аутентифікації користувачам, подальша обробка інформації для обчислення мінімального питомого повітрообміну та обчислення питомої потужності вентиляційного обладнання. З огляду на

функції сервера нижнього рівня вибрано його апаратне та програмне забезпечення (Таблиця 3.3).

Таблиця 3.3 – Конфігурація сервера

Комплектуючі	Модель
Набір мікросхем	Intel 5100
Процесори	1 або 2 Intel Xeon 5xxx (до 8 ядер)
Швидкість системної шини	1333 МГц
Максимальний обсяг оперативної пам'яті	16 Гбайт двоканальної DDRII-667 ECC
Слоти розширення	1xPCI-E 16x, 1xPCI-E 8x
Вбудовані контролери	2 порти LSI SAS 1064
Оптичні накопичувачі	DVD CD-RW
Опціональні контролери	Контролери SAS RAID з підтримкою BBU, адаптери FibreChannel, 10G Ethernet и InfiniBand HCA
Максимальна кількість дисків	2xSAS 3.5" чи 4xSATA/SSD 2.5" з можливістю гарячої заміни
Ємність дискової підсистеми	До 600 Гбайт SAS/ 2 Тбайт SATA
Мережні інтерфейси	2x Intel Gigabit Ethernet
Відеоконтролер	ASPEED AST2000, 8 Мбайт
Інтерфейси	Задня панель: VGA, RS232, 3xRJ-45, 2xUSB, 2xPS2; передня панель: 2xUSB
Керування системою	IPMI 2.0. KVM over IP, Virtual Media, Ethernet
Підтримувані ОС	SuSE Linux Enterprise Server 10; Novell Open Enterprise Server; Семейство Microsoft Windows Server 200X; Red Hat Enterprise Linux 5.0; Sun Solaris 10

Продовження таблиці 3.3

Розміри	(Ш x В x Д), мм 1U, 510 x 430 x 44 мм (глибина стійки не менше 800 мм)
Блок живлення	400 Вт

3.2.3 Реалізація принципової схеми комп'ютерної системи на санітарно-пропускному пункті

На основі обраної елементної бази, опису входів та виходів системи реалізована принципова схема об'єкту, на рисунку 3.6

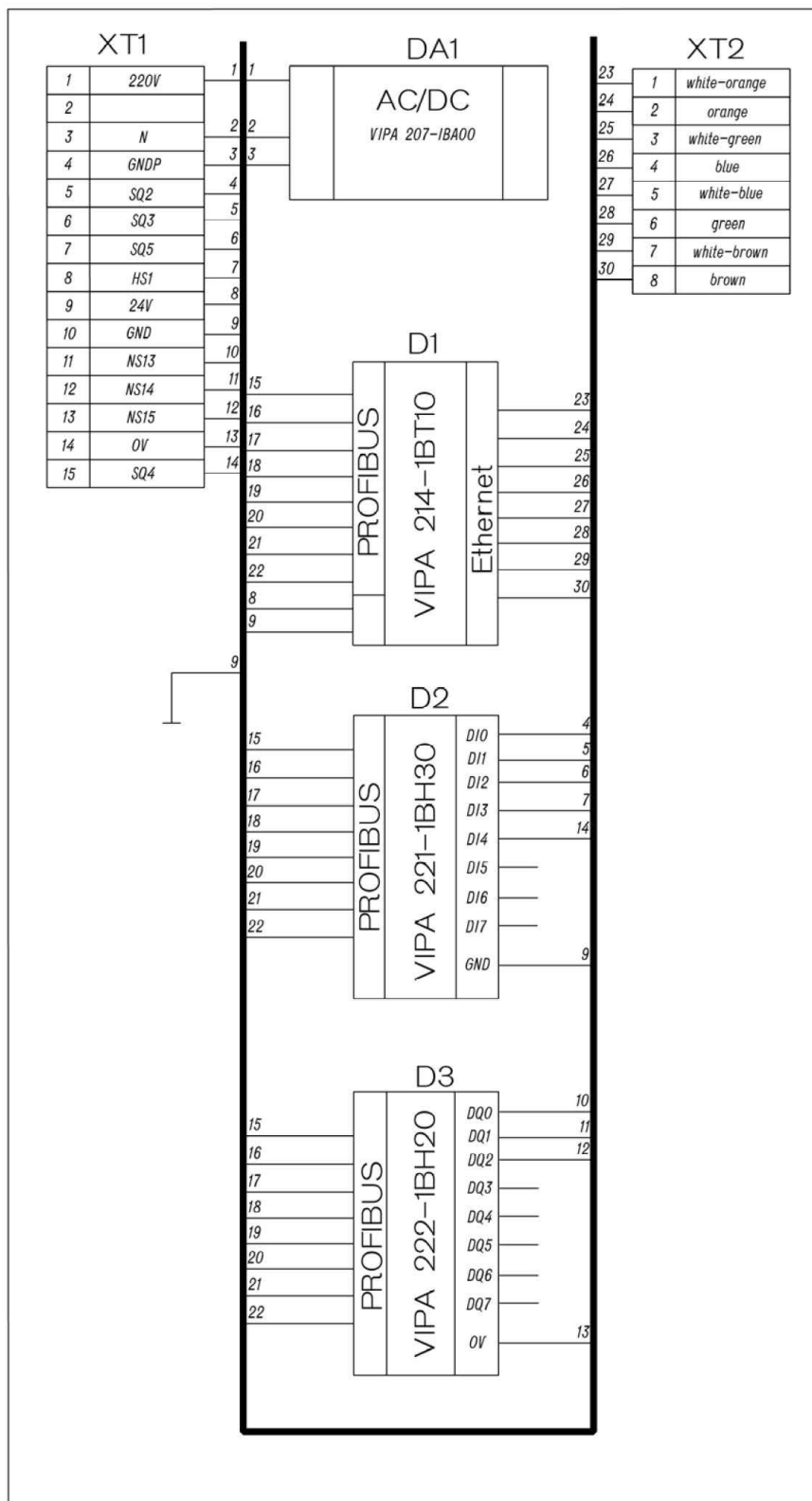


Рисунок 3.6 – Принципова схема комп'ютерної системи на санітарно-пропускному пункті

Отримана принципова схема може бути використана для розробки математичних моделей керування комп'ютерною системою на санітарно-пропускному пункті.

ВИСНОВОК

На основі обзору технологічного процесу комп'ютерної системи на санітарно-пропускному пункті було побудовано функціональну та принципову схему керування, що в подальшому будуть використані для розробки програмного забезпечення проектованої комп'ютерної системи.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Призначення та сфера використання програмного забезпечення

Програмне забезпечення призначене збирати, систематизувати та порозводити розрахунки на основі даних від санітарно-пропускних пунктів, для регулювання потужності вентиляційної системи на харчовому м'ясообробному виробництві.

Серверна частина, детальну розробка якої проводиться у рамках даної кваліфікаційної роботи є інтеграційною частиною двох комп'ютерних систем: системи на санітарно-пропускному пункті та вентиляційної.

4.2 Обґрунтування технічних характеристик програми

Програма розташована на сервері бази даних комп'ютерної системи на санітарно-пропускному пункті повинна виконувати наступні функції:

- Проведення аутентифікації працівників;
- Збір та обробку даних про пересування робітників виробництва через санітарно-пропускні пункти;
- Розрахунок мінімального питомого повітрообміну;
- Регулювання потужності витяжної системи на основі розрахованого повітрообміну;
- Внесення змін до бази даних працівників, що можуть відвідувати санітарні зони.

Програма не використовує безпосереднє керування апаратним обладнанням об'єктів та може бути інтегрована у будь яку систему, що передбачає подібну інтеграцію.

Проектована програма повинна використовувати такі вхідні дані:

- Пакети даних від картридера, що містять у собі дані від сканованої карти;

- Дискретні сигнали від контролерів санітарно-пропускних пунктів, що звітують про завершення проходження санітарного контролю працівником;

Програма повинна на основі вхідних даних повинна видавати такі вихідні дані:

- Дискретні сигнали до контролерів, що дозволяють або забороняють відвідування працівником даної зони;
- По запиті ЛПР програма має формувати таблицю відвідувачів, що містить в собі дані про працівників, час їх перебування на території санітарної зони;
- Дані про необхідну потужність вентиляційної системи, що забезпечить мінімальний питомий повітрообмін у приміщенні.

Технічні та програмні засоби для розробки програми включають комп'ютер, що підтримує середовище розробки IDLE Python, мова програмування Python.

4.3 Опис розробленої програми

4.3.1 Загальні відомості

Програма розташована на сервері, за допомогою облікового запису адміністратора ЛПР має можливість вносити зміни до бази даних робітників з будь-якого пристрою, що знаходиться у мережі підприємства. Файли конфігурацій винесені окремо, що спрощує первісне налаштування системи. Мова програмування, що використовувалась для написання програми – Python.

4.3.2 Функціональне призначення

Програма здійснює збір та обробку інформації про проходження працівниками санітарно-пропускних пунктів та виконує розрахунок мінімального питомого повітрообміну та витрати електроенергії на його забезпечення з урахуванням кількості працівників у приміщенні. Для наочності,

програма також будує графік витрат електроенергії на забезпечення мінімального питомого повітрообміну. Програма не керує безпосередньо санітарно-пропускним пунктом, його програма реалізована у контролері. Програма, також не керує вентиляційним обладнанням, а лише видає параметри, що необхідні для регулювання його роботи з метою зменшення витрат на електроенергію.

4.3.3 Опис логічної структури програми

Програма має наступну структуру:

- Текстовий документ `requirements.txt`, що включає в себе обов'язкові для установки імена бібліотек, без яких робота програмного забезпечення у будь-якому середовищі програмування неможлива. Усі найменування бібліотек мають формат строки, додатково можуть включати у себе версію, записані нижнім регістром та не містять пробілів. Завантаження зазначених бібліотек виконується за допомогою попередньо завантаженої програми «`pip`», команда має наступний вигляд:

$$\text{pip install -r path_to_txt} \quad (4.1)$$

- 1) флаг «`r`» вказує програмі, що зараз буде зчитаний список бібліотек з текстового файлу;
 - 2) `path_to_txt` – повний шлях до файлу `requirements.txt`, що містить найменування потрібних для роботи програми бібліотек.
- Точка входу «`main.py`», відповідає за запуск клієнт-серверної частини усього технічного рішення. Даний модуль реалізує як серверну, так і клієнтську частини. Під серверною частиною треба розуміти: зберігання інформації; обробку інформації; реалізацію прийому-передачі інформації в умовному вигляді, в якому його зможе прийняти інша частина розробленої системи. Під клієнтською частиною треба розуміти: обробку інформації браузером (кінцевим користувачем та/або його програмним забезпеченням); прийомом інформації, інтерпретацією та/або

представленням інформації у задовольняючому кінцевого користувача вигляді.

- Файл конфігурації клієнт-серверної частини реалізованої системи – «main.ini». У файлі конфігурацій зберігаються дані, необхідні та достатні для роботи модуля, що відповідає за серверну частину реалізованого комплексу. Даний файл може містити:
 - 1) Адрес локальної мережі вузла, що приймає зумовлену інформацію (у нашому випадку енергоспоживання), в форматі IPv4;
 - 2) Відповідний порт сокетного з'єднання, на якому в подальшому буде заходитися мережевий процес обробки інформації;
 - 3) Мінімальний повітрообмін на одну людину при даних умовах у форматі цілого числа.
- Директорія «templates», у якій знаходяться наступні файли:
 - 1) Index.html – файл розмітки, що відображається браузером (кінцевим користувачем та/або його програмним забезпеченням) за адресою вузла, на якому розгорнуто серверну частину, із відповідним портом 5000 та без додаткової вказівки на підсторінку. Відображає переходи на усі можливі сторінки.
 - 2) Worker.html – файл розмітки, що відображається браузером (кінцевим користувачем та/або його програмним забезпеченням) за адресою вузла, на якому розгорнуто серверну частину, із відповідним портом 5000 у підсторінці /<i>, де і дорівнює цілочисельному значенню, що відповідає порядковому номеру у базі даних персоналу. Відображає інформацію про конкретного працівника.
 - 3) Worker_table.html - файл розмітки, що відображається браузером (кінцевим користувачем та/або його програмним забезпеченням) за адресою вузла, на якому розгорнуто серверну частину, із відповідним портом 5000 у підсторінці /all. Відображує список

персоналу, що допущений до перебування на території санітарної зони; мінімальний питомий повітрообмін та графік споживання електроенергії на забезпечення питомого повітрообміну.

4.4 Очікувані техніко-економічні показники

Використання комп'ютерної системи на санітарно-пропускному пункті здатне зменшити витрати на електроенергію при забезпеченні мінімального питомого повітрообміну вентиляційною системою мінімум на 18000 грн. на рік. Ця сума є актуальною при введенні системи лише в одному цеху, що налічує 100 працівників, та при показнику $SFP=0.55$, тобто вентиляційна система має високий ККД. На великому підприємстві економічний ефект пропорційно збільшиться.

ВИСНОВОК

Для рішення задачі по збору та систематизації даних, що надходять від санітарно-пропускних пунктів; розрахунку мінімального питомого повітрообміну та витрати електроенергії, було визначено призначення та сферу використання системи та обґрунтування її технічних характеристик, що дало можливість скласти технічне завдання для програми.

Як результат, було розроблено програму на мові програмування Python, що відповідає всім умовам технічного завдання на програмне забезпечення. А також для наочності було застосовано: мову програмування SQL, для створення тестової бази даних та мову програмування HTML, для оформлення клієнтської частини програмного забезпечення.

5 ЕКСПЕРИМЕНТАЛЬНИЙ РОЗДІЛ

5.1 Формулювання вимог до експерименту

Необхідно експериментальним шляхом перевірити працездатність розробленої у ході кваліфікаційної роботи комп'ютерної системи на санітарно-пропускному пункті.

Для проведення експерименту необхідно провести випробування розробленої програми.

Мета експерименту – дослідним шляхом визначити працездатність та відмовостійкість системи відстеження пересувань працівників через пункти санітарного контролю.

5.2 Опис експерименту

Для проведення експерименту, використаємо комп'ютер, маршрутизатор, програмне забезпечення IDLE Python, браузер Google Chrome, MySQL Workbench, аналізатор трафіку Wireshark та розробимо функцію, що імітує сигнали від картридерів санітарно-пропускних пунктів.

Суть експерименту в тому, щоб подаючи сигнали про проходження санітарно-пропускних пунктів із різноманітною періодичністю, спостерігати зміну графіку витрати електроенергії, мінімальний питомий повітрообмін та зміни у таблиці пересувань працівників. Також потребує підтвердження здатність програми передавати розраховані параметри відповідній системі.

Оскільки будуть проводитись дослідження програмного забезпечення ДСТУ 19.301-79 передбачає написання контрольного прикладу для проведення випробувань.

5.3 Контрольний приклад для проведення випробувань

5.3.1 Підготовка до роботи

Таблиця 5.1 – Послідовність дій при підготовці до роботи та запуску

№ з/п	Дія	Результат
1.	Перевірити цілісність дротів та з'єднань, наявність напруги 220В	Цілісність не порушена, напруга живлення в мережі знаходиться в діапазоні від 200 до 250В
2.	Ввімкнути комп'ютер	Комп'ютер ввімкнувся, завантажилась операційна система Windows XP або Windows 7
3.	Завантажити IDLE Python 3.7+ та відкрити у ньому проект SPP	На екрані відкрито середовище IDLE Python 3.7+ із завантаженим проектом SPP
4.	Відкрити браузер Google Chrome та ввести до адресної строки <code>http://0.0.0.0:5000</code>	У браузері відобразилось вікно аутентифікації користувача (Рисунок 5.1)
5.	Ввести у поле Логін admin, та у поле Пароль 12345	У браузері відобразилась головна сторінка інтерфейсу користувача (Рисунок 5.2)

5.3.2 Дослідження працездатності та відмовостійкості програми

Таблиця 5.2 – Послідовність дій для дослідження працездатності програми

№ з/п	Дія	Результат
1.	Розкоментувати функцію імітації пересування працівників	Нічого не відбувається, база даних працівників пуста
2.	Скопіювати зміст файлу «тест» до файлу db_sqlite3, натиснути Save.	Через 5 секунд з'явиться таблиця співробітників та вони почнуть заповнювати приміщення відповідно до логіки функції імітації.

Продовження таблиці 5.2

3.	Натиснути кнопку «View»	Здійснюється перехід на іншу сторінку (Рисунок 5.3), де відображено клікабельні кнопки, з іменами працівників.
4.	Натиснути на будь-яке ім'я	Здійснюється перехід на іншу сторінку, на сторінці відображені усі дані про працівника. (Рисунок 5.4)
5.	Натиснути кнопку «View all staff»	Здійснюється перехід на головну сторінку.
6.	Прогорнути колесо миші донизу.	Внизу сторінки (Рисунок 5.5) ми бачимо поле «Мінімальний повітрообмін», де відображено мінімальний повітрообмін при поточній кількості працівників, а також графік споживання електроенергії.
7.	Ввести у Wireshark команду <code>tcpdump -i wlan0 port 8888</code>	Аналіз трафіку через Wireshark показує, що сервер через порт <code>eth0</code> відсилає пакет з даними раз на хвилину. Його також можна відстежити по пункту призначення. Пункт призначення та порт можна змінити у файлі <code>main.config</code> . (Рисунок 5.6)

Продовження таблиці 5.2

8.	Перевірити відповідність розрахованих параметрів.	Розрахована величина повітрообміну відповідає мінімальному питомому повітрообміну на кількість осіб у приміщенні, розташування останньої крапки на графіку споживання електроенергії також відповідає розрахункам
----	---	---

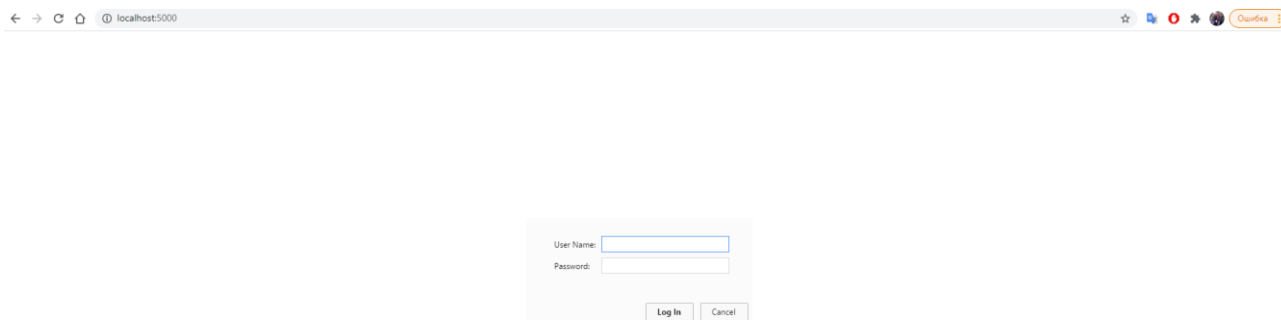


Рисунок 5.1 – Вікно аутентифікації

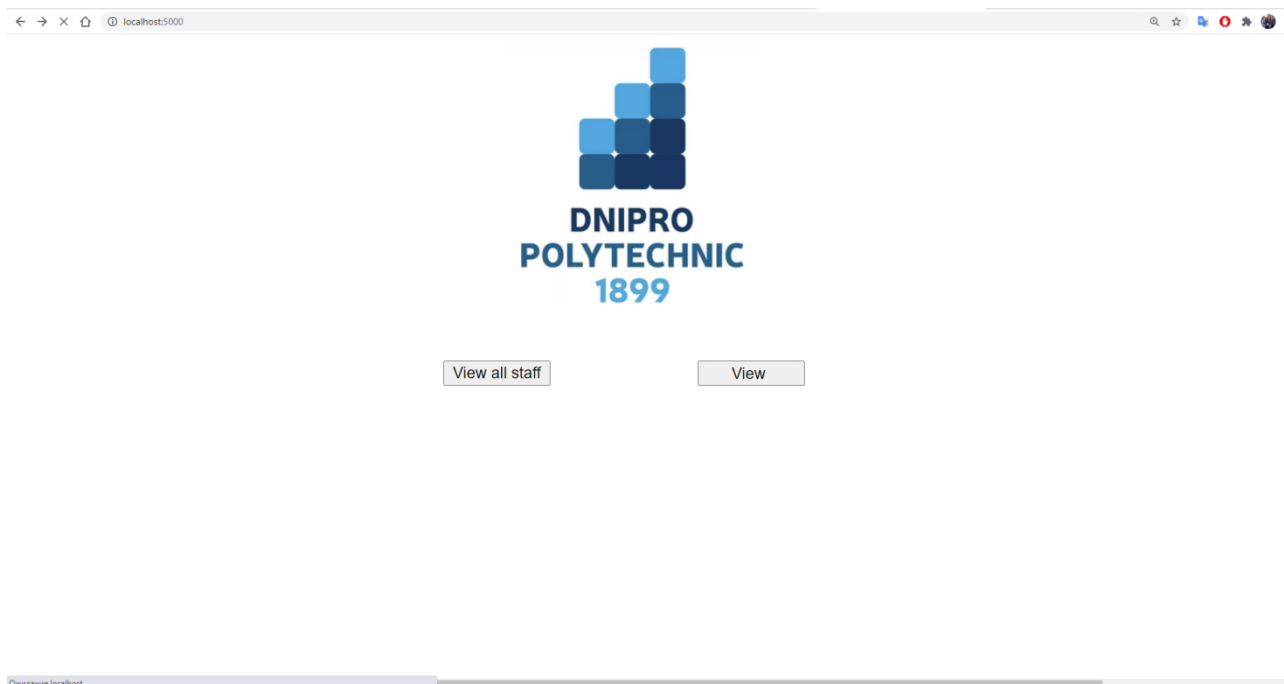


Рисунок 5.2 – Головна сторінка інтерфейсу користувача



Рисунок 5.3 – Сторінка вибору працівника



Рисунок 5.4 – Інформація про працівника



Рисунок 5.5 – Сторінка з відображенням поточних параметрів системи

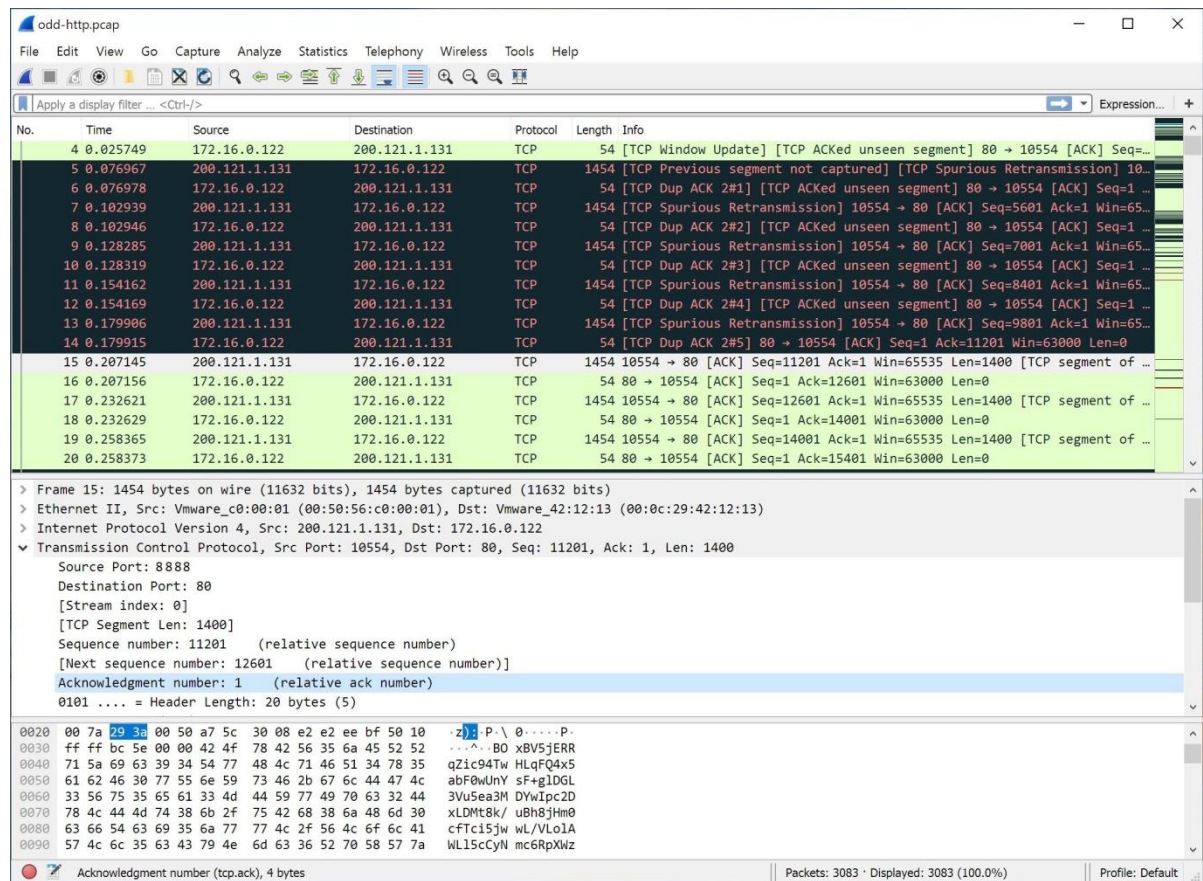


Рисунок 5.6 – Дамп трафіку сервера

Перевірка розрахунків:

$$M = N \times w, 73 \times 60 = 4380 \quad (5.1)$$

$$W = M \times SFP, 4380 \times 0.55 = 2409 \quad (5.2)$$

- W – витрата електроенергії (Вт);
- w – мінімальний повітрообмін на людину ($\text{м}^3/\text{год}$);
- N – кількість людей у приміщенні;
- SFP – витрата електроенергії на виробництво 1м^3 повітря за годину ($\text{Вт}/\text{м}^3/\text{год}$);
- M – мінімальний питомий повітрообмін приміщення.

5.4 Аналіз результатів експерименту

У результаті виконання контрольного прикладу було виявлено повну відповідність отриманих результатів заявленим.

Відповідність пункту 5.1 свідчить про справність програмно-апаратного комплексу.

Відповідність пункту 5.2 свідчить про адекватну роботу програми, а саме:

- Вхід до санітарної зони припустимий тільки для працівників, що занесені до бази даних комп'ютерної системи;
- Програма розраховує необхідні параметри виходячи з даних, що надходять від санітарно-пропускних пунктів;
- Графік споживання електроенергії наочно демонструє економічну вигоду від використання системи;
- Аналіз трафіку, з допомогою Wireshark, підтвердив працездатність модулю комунікації із системою, що використовує розраховані параметри питомого повітрообміну та необхідної потужності обладнання для його забезпечення;
- Результати одержані у ході роботи програми, збігаються з результатами, що одержані шляхом підстановки величини кількості працівників у приміщенні до формул (5.1) та (5.2)

ВИСНОВОК

У результаті дослідження було підтверджено доцільність розгляду кількості працівників у приміщенні, як випадкової величини з нелінійним розподілом. Тестова функція імітації пересування працівників, дозволила перевірити функціонал розробленої програми та підтвердити її відповідність умовам викладеним у технічному завданні на розробку програмного забезпечення. Аналіз трафіку, за допомогою програмного забезпечення Wireshark, дозволив виділити та проаналізувати пакети з даними, що з однаковою періодичністю сервер відсилає системі керування вентиляційним обладнанням. Одержані величини мінімального повітрообміну приміщення та споживання електроенергії співпадають з розрахованими.

ВИСНОВКИ

У ході виконання магістерської кваліфікаційної роботи виведена залежність між кількістю людей у приміщенні та питомою потужністю вентиляційної системи, що забезпечує мінімальний питомий повітрообмін.

Дослідження стану питання, дозволило довести доцільність розробки комп'ютерної системи на санітарно-пропускному пункті, базуючись на даних, про роботу санітарно-пропускних пунктів [], та вимог до витяжних механічних систем на харчових підприємствах [].

Представлення кількості працівників у приміщенні, як випадкової величини з нелінійним розподілом дозволило математично довести доцільність впровадження комп'ютерної системи на санітарно-пропускному пункті.

Також, шляхом аналізу характеристик стандартних дезінфікуючих засобів, було сформовано вимогу використання на санітарно-пропускних пунктах розчинів на основі четверткових амонієвих солей, зумовлену їх високою проникною здатністю.

На основі обзору технологічного процесу комп'ютерної системи на санітарно-пропускному пункті було побудовано функціональну та принципову схему керування, що були використані для розробки програмного забезпечення проектованої комп'ютерної системи.

Сформована концепція, узагальнююча вище перелічені результати, дозволила створити програмне забезпечення, що відповідає вимогам до складу магістерської роботи.

Проведення експерименту дозволило підтвердити відповідність розробленої системи вимогам, що були сформульовані у технологічному завданні.

ПЕРЕЛІК ПОСИЛАНЬ

1. Дипломування. Методичні рекомендації до виконання кваліфікаційної роботи магістра студентами галузь знань 12 Інформаційні технології спеціальності 123 Комп'ютерна інженерія / Л.І. Цвіркун, С.М. Ткаченко; Міністерство освіти і науки України, Національний технічний університет «Дніпровська політехніка». – Дніпро: НТУ «ДП», 2019. – 41с.
2. Системы качества. Управление качеством пищевых продуктов на основе принципов ХАССП / ВНИИС/ 2009. – 12с.
3. Производственная санитария на пищевых производствах [Електронний документ] режим доступу <http://tridar.com.ua/stati/stati/proizvodstvennaya-sanitariya-na-pishchevykh-predpriyatiyakh/>
4. Применение дезинфицирующих средств: обзор [Електронний документ] режим доступу <https://www.chemitech.ru/blog/article/primenenie-dezinfitsiruyushchikh-sredstv-obzor/>
5. Инструкция по санитарной обработке технологического оборудования и производственных помещений на предприятиях мясной промышленности / Ю.Г. Костенко, В.О. Риболовський; ВНИИ мясной промышленности им. В.М. Горбатова. 2003. – 177 с.
6. Определение необходимости воздухообмена помещений. Рекомендации к проектированию [Електронний документ] режим доступу <https://vents.ua/opredelenie-neobhodimosti-vozduhoobmena-pomesenij-rekomendacii-k-proektirovaniu>
7. Вентиляция, кондиционирование и очистка воздуха на предприятиях пищевой промышленности / Е.А. Штокман; М.АСВ. 2001. - 564с.
8. Теорія систем. Методичні рекомендації до виконання лабораторних робіт для студентів напряму підготовки 151 «Автоматизація та комп'ютерно-інтегровані

технології» - Автори: М.М. Трипутень, В.Є. Кириченко, Є.К. Воскобойник;
М-во освіти і науки України, Нац. Гірн. Ун-т. – Д.:НГУ, 2017.-36с.

ДОДАТОК А

Технічне завдання

Програмне забезпечення комп'ютерної системи на санітарно-пропускному
пункті

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

**ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ НА
САНІТАРНО-ПРОПУСКНОМУ ПУНКТІ**

**Технічне завдання
804.02070743.20023-01 1201**

Аркушів 4

2020

ВСТУП

Найменування програми: «Програмне забезпечення сервера даних на санітарно-пропускному пункті».

Коротка характеристика сфери застосування: програмне забезпечення призначене для серверу бази даних, що на основі даних від санітарно-пропускних пунктів, розраховує необхідний повітрообмін для вентиляційної системи на харчових виробництвах.

1 ПІДСТАВИ ДЛЯ РОЗРОБКИ

Наказ про дипломне проектування НТУ «Дніпровська політехніка» №888с, від 22.10.2020.

5 ПРИЗНАЧЕННЯ РОЗРОБКИ

Для моделювання та виконання розрахунків мінімального повітрообміну, на основі даних від санітарно-пропускних пунктів, на харчовому м'ясообробному виробництві в кваліфікаційній роботі розроблено програмне забезпечення, яке покликане забезпечити точні та достовірні результати експериментів з реалізацією всіх необхідних функцій.

3 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Вимоги до функціональних характеристик

Програма повинна забезпечувати реалізацію наступних функцій системи:

- Контроль доступу працівників до санітарних зон підприємства;
- Дезінфекція та очистка рук та підошов взуття працівників;
- Збір, обробку та занесення до бази даних інформації, про проходження робітниками підприємства санітарно-пропускних пунктів;
- Розрахунок мінімального питомого повітрообміну у приміщенні на основі даних про пересування працівників;

- Розрахунок потужності вентиляційної системи відповідно до інформації про кількість працівників, що знаходяться у приміщенні.

3.3 Умови експлуатації

Копія програмного забезпечення повинна зберігатися на CD-диску, для якого повинні дотримуватися такі умови експлуатації:

- температура навколишнього середовища не вище 28°C;
- вологість повітря не вище 80%;
- захист від нагрівання і попадання прямих сонячних променів.

Для обслуговування та контролю роботи даного програмного забезпечення потрібен фахівець, що має навички роботи з базами даних, та ознайомлений з принципами роботи цієї системи та експлуатаційними вимогами.

3.4 Вимоги до складу і параметрів технічних засобів

Для розробки програми, потрібен персональний комп'ютер із наступними технічними характеристиками:

- Процесор 2.6 ГГц і вище;
- Не менше 10 Гбайт ОЗУ;
- Не менше 500 Мбайт вільного дискового простору.

3.5 Вимоги до інформаційної та програмної сумісності

Для коректної роботи, налагодження та супроводу програми, система повинна мати таке програмне забезпечення: ОС Windows 8/8.1/10, або ОС Linux; IDLE Python версії 3.7 і вище.

4 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

Попередній склад програмної документації:

- Специфікація;
- Текст (схема) програми.

5 ПОРЯДОК КОНТРОЛЮ І ПРИЙМАННЯ

Програмне забезпечення комп'ютерної системи на санітарно-пропускному пункті, повинно пройти тестування для подальшого застосування на технологічному об'єкті. Після первісного налаштування необхідно за допомогою тестової функції та бази даних упевнитися у коректності роботи програми.

ДОДАТОК Б

Специфікація

Програмне забезпечення комп'ютерної системи на санітарно-пропускному пункті

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

**ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ НА
САНІТАРНО-ПРОПУСКНОМУ ПУНКТІ**

Специфікація

804.02070743.20023-01 1201

Аркушів 2

2020

Позначення	Найменування	Примітка
	Документи	
804.02070743.20023-01 1201	Програмне забезпечення комп'ютерної системи на санітарно-пропускному пункті	Детальна розробка серверу бази даних

ДОДАТОК В

Текст програми

Програмне забезпечення комп'ютерної системи на санітарно-пропускному
пункті

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

**ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ НА
САНІТАРНО-ПРОПУСКНОМУ ПУНКТІ**

Текст програми

804.02070743.20023-01 1201

Аркушів 24

2020

АНОТАЦІЯ

У документі наведено текст програми, розробленої у кваліфікаційній роботі. До тексту програми додано короткі коментарі.

Розроблена програма є реалізацією серверної частини комп'ютерної системи на санітарно-пропускному пункті для робочого та тестового режимів.

Програма написана на мові Python у середовищі розробки IDLE Python 3.7.9.

ЗМІСТ

1. Функція main	4
2. Функція імітації пересувань працівників	5
3. Ініціалізація бази даних	8
4. Формування бази даних	8
5. Стили CSS	17
6. Шаблони відображення	18
7. Файл налаштувань	24

Текст програми мовою Python

```
#main Python File
```

```
import os
```

```
import sys
```

```
os.chdir(os.path.dirname(os.path.abspath(__file__)))
```

```
sys.path.append(os.path.dirname(os.path.abspath(__file__)))
```

```
import time
```

```
from faker import Faker
```

```
from flask_cors import CORS
```

```
from random import randrange
```

```
from threading import Thread
```

```
from socket import socket, AF_INET
```

```
from configparser import ConfigParser
```

```
from flask import Flask, render_template, request, jsonify
```

```
from lib import Database
```

```
from lib.configs import MainConfig
```

```
app = Flask(__name__)
```

```
CORS(app)
```

```
db = Database(); db.recreate_staff_table()
```

```
application_config = MainConfig()
```

```
fake = Faker()
```

```
DESTINATION_SOCKET = (application_config.host, int(application_config.port))
```

```
MIN_HUMAN_CONSUMPTION = float(application_config.consumption) # m^3/ч
ENERGY_CONSUMPTION_PER_M3 = 0.55 # Коэффициент потребления
электроэнергии

GLOBAL_CONSUMPTION_VALUE = 0
GLOBAL_ENERGY_CONSUMPTION = 0
PREVIOUS_ENERGY_CONSUMPTIONS = []

MAX_PEOPLE = 100
PERCENT = 0.1

def logic():
    """
    Функция просчета искомых значений и отправки последних на указанный в
    настройках сервер
    :return:
    """
    global GLOBAL_CONSUMPTION_VALUE,
    GLOBAL_ENERGY_CONSUMPTION

    target = socket(AF_INET)

    upper_downer = True # Функция имитации перемещения сотрудников
    duplicate = True

    while True:
        try:
            working_human_counter = 0
            workers = db.get_all_workers()
```

```

if not(workers):
    db.add_worker(fake.name())

    if len(workers) <= MAX_PEOPLE + int(PERCENT * MAX_PEOPLE) and
upper_downer:
        db.add_worker(fake.name())

    if len(workers) == MAX_PEOPLE + int(PERCENT * MAX_PEOPLE) and
upper_downer:
        print('Changing upper_downer')
        upper_downer = False

    if len(workers) == int(MAX_PEOPLE * 0.2) and duplicate and
not(upper_downer):

PREVIOUS_ENERGY_CONSUMPTIONS.extend([PREVIOUS_ENERGY_CONS
UMPTIONS[-1]] * int(MAX_PEOPLE / 5))

    while len(workers) < MAX_PEOPLE + int(PERCENT * MAX_PEOPLE)
- 1:
        db.add_worker(fake.name())
        workers = db.get_all_workers()
        PREVIOUS_ENERGY_CONSUMPTIONS.append(
            int(MIN_HUMAN_CONSUMPTION * len(workers) *
ENERGY_CONSUMPTION_PER_M3) if len(workers) <= MAX_PEOPLE else
            int(MIN_HUMAN_CONSUMPTION * MAX_PEOPLE *
ENERGY_CONSUMPTION_PER_M3)
        )
        duplicate = False

    if len(workers) > 1 and not(upper_downer):

```

```

    print('RES', db.delete_last()) # Конец функции имитации перемещения
    сотрудников

```

```

    for w in workers:

```

```

        if w['working']:

```

```

            working_human_counter += 1

```

```

            GLOBAL_CONSUMPTION_VALUE =
            int(MIN_HUMAN_CONSUMPTION * working_human_counter)

```

```

            GLOBAL_ENERGY_CONSUMPTION =
            int(GLOBAL_CONSUMPTION_VALUE * ENERGY_CONSUMPTION_PER_M3)

```

```

            PREVIOUS_ENERGY_CONSUMPTIONS.append(

```

```

                GLOBAL_ENERGY_CONSUMPTION if len(workers) <=
                MAX_PEOPLE else

```

```

                int(MIN_HUMAN_CONSUMPTION * MAX_PEOPLE *
                ENERGY_CONSUMPTION_PER_M3)

```

```

            )

```

```

        except Exception as e:

```

```

            print('HERE', e)

```

```

    try:

```

```

        target.connect(DESTINATION_SOCKET)

```

```

        target.send(

```

```

            bytes([GLOBAL_ENERGY_CONSUMPTION,
            GLOBAL_CONSUMPTION_VALUE])

```

```

        )

```

```

    except Exception as e:

```

```

        pass # print(e)

```

```

# time.sleep(0.01) # In seconds

@app.route("/", methods=['GET'])
@app.route('/<identify>')
def workers(identify=None):
    """
    Обработка запросов к сайту
    :param identify: string, do not use
    :return: Обработанную страницу
    """
    global GLOBAL_CONSUMPTION_VALUE,
    GLOBAL_ENERGY_CONSUMPTION

    worker = None

    if identify and identify != 'all':
        worker = db.get_worker_by_id(identify)
    elif identify == 'all':
        worker = db.get_all_workers()

    if isinstance(worker, dict):
        return render_template(
            'worker.html',
            name=worker.get('name', 'UNDEFINED'),
            start_time=time.ctime(worker.get('start_time', 0)) if worker.get('working', 0)
else "",
            working="YES" if worker.get('working', 0) else "NO"
        )
    elif isinstance(worker, list):

```

```

for w in worker:
    w['working'] = "YES" if w.get('working', 0) else "NO"
    w['start_time'] = time.ctime(w.get('start_time', 0)) if w.get('working', 0) else
"""

return render_template(
    'worker_table.html',
    workers=worker,
    value=GLOBAL_CONSUMPTION_VALUE,
    energy=PREVIOUS_ENERGY_CONSUMPTIONS
)

return render_template('index.html', workers=db.get_all_workers())

if __name__ == '__main__':
    # Thread(target=workers_updater, args=(), daemon=True).start()
    Thread(target=logic, args=(), daemon=True).start()
    app.run(host='0.0.0.0', port=5000)

#main Phyton File
#lib/database
from .database import Database
#lib/database/init
#lib/database/database
"""

SQLite3 Database object

```

```
~~~~~
```

Defines singleton database object to work with it on app level.

```
"""
```

```
__all__ = ['Database']
```

```
__version__ = '0.1.0'
```

```
import os
```

```
import sqlite3
```

```
from time import time
```

```
DATABASE_NAME = 'db_.sqlite3'
```

```
class Singleton(type):
```

```
    """
```

```
    Singleton class for making only one connection to database
```

```
    """
```

```
    _instances = { }
```

```
    def __call__(cls, *args, **kwargs):
```

```
        if cls not in cls._instances:
```

```
            cls._instances[cls] = super().__call__(*args, **kwargs)
```

```
        return cls._instances[cls]
```

```
class Database(metaclass=Singleton):
```

```
    """
```

```
    Class for database. Contains all methods for operating with staff-table
```



```

"""
__connection = None

def __init__(self):
    self.connect()

def connect(self):
    """
    Connect to database db.sqlite. Create new database if file not exist.
    """
    if self.__connection is None:
        try:
            self.__connection = sqlite3.connect(
f'file:{os.path.dirname(os.path.abspath(__file__))}/{DATABASE_NAME}?mode=r
w',
                check_same_thread=False,
                uri=True)
        except sqlite3.OperationalError:
            self.__create_database()
            self.__connection.row_factory = sqlite3.Row

def create_staff_table(self):
    """
    Create staff table. Contains primary id, name, working and start_time.
    :return: None
    """
    self.__connection.cursor().execute(

```

```

"CREATE TABLE staff ("
    "id INTEGER PRIMARY KEY,"
    "name TEXT,"
    "working BOOLEAN DEFAULT TRUE,"
    "start_time REAL NOT NULL)"
)
self.__connection.commit()

```

```

def recreate_staff_table(self):
    """
    Recreate empty staff-table
    :return:
    """
    cursor = self.__connection.cursor()
    try:
        cursor.execute('DROP TABLE staff')
        self.__connection.commit()
    except Exception as e:
        print(e)
    self.create_staff_table()

```

```

def add_worker(self, name=None):
    """
    Adds worker to staff-table
    :param name: string, name of worker
    :return:
    """

```

```

if name:
    cursor = self.__connection.cursor()
    cursor.execute(f'INSERT INTO staff VALUES (NULL, "{name}",
TRUE, "{time()}")')
    self.__connection.commit()
    return cursor.lastrowid
return None

```

```

def get_worker_by_id(self, i):
    """
    Fetch only one worker by id from database.
    """
    cursor = self.__connection.cursor()
    res = cursor.execute(f'SELECT * FROM staff WHERE id = {i}')
    worker = res.fetchone()
    if worker:
        return dict(worker)
    else:
        return None

```

```

def delete_worker_by_id(self, i):
    """
    Delete worker from staff-table via primary id
    :param i: int, id
    :return:
    """
    if i:

```

```

cursor = self.__connection.cursor()
    cursor.execute(f'DELETE FROM staff WHERE id={i}')
    self.__connection.commit()
    return cursor.lastrowid
return None

```

```
def get_all_workers(self):
```

```
    """
```

```
    Returns all workers in staff-table, if they are, otherwise returns None
```

```
    :return: dict
```

```
    """
```

```

cursor = self.__connection.cursor()
res = cursor.execute('SELECT * FROM staff')
workers = res.fetchall()
if workers:
    return [dict(i) for i in workers]
else:
    return None

```

```
def update_worker(self, ident=-1, check=False):
```

```
    """
```

```
    Updates worker via id in staff-table, puts check-flag to working and new time, if
check-flag
```

```
    :param ident: primary id, int
```

```
    :param check: bool
```

```
    :return:
```

```
    """
```

```

if ident > 0:
    cursor = self.__connection.cursor()
    if check:
        cursor.execute(f'UPDATE staff SET '
                        f'working = {check}, '
                        f'start_time = "{time()}" '
                        f'WHERE id = {ident}')
    else:
        cursor.execute(f'UPDATE staff SET '
                        f'working = {check} '
                        f'WHERE id = {ident}')

    self.__connection.commit()

def __create_database(self):
    """
    Create db.sqlite file with tables and initial settings.
    """
    self.__connection = sqlite3.connect(DATABASE_NAME,
    check_same_thread=False)

    self.create_staff_table()

if __name__ == '__main__':
    db = Database()
    db.recreate_staff_table()
#lib/database/database

```

```

#lib/init
from .database import Database
#lib/init
#lib/configs
from abc import ABC
from configparser import ConfigParser

class Config(ABC):
    """
    Abstract config class
    Encapsulation working with configs
    """

    def __setattr__(self, key: str, value: str):
        """
        Setter for update option configs
        Not create new option if not exist, only update existing

        :param key: str, option name
        :param value: str, data
        :return: None
        """

        # exception if type not string
        if type(value) != str:
            raise TypeError('type must be str')

```

```

config = ConfigParser()
config.read(self.path)

# exception if not exist
__ = config[self.section][key]

config.set(self.section, key, value)

with open(self.path, 'w') as config_file:
    config.write(config_file)

def __getattr__(self, key: str) -> str:
    """
    Return option data

    :param key: str, option name
    :return: str, data from this option
    """

    config = ConfigParser()
    config.read(self.path)

    return config[self.section][key]

def get_keys(self) -> list:
    """
    Getting all keys from selected config

```

```
:return: list, collection keys from config
```

```
"""
```

```
config = ConfigParser()
```

```
config.read(self.path)
```

```
return [item[0] for item in config.items(self.section)]
```

```
class MainConfig(Config):
```

```
    path = 'main.ini'
```

```
    section = 'APP'
```

```
#lib/configs
```

```
#static/css/worker
```

```
body {
```

```
    //background: #202020;
```

```
}
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    padding: 5px;
```

```
}
```

```
table {
```

```
    width: 25%;
```

```
    border-spacing: 15px;
```

```
    margin-left: auto;
```



```
    margin-right: auto;
}
```

```
h2 {
    text-align: center;
}
```

```
p {
    text-align: center;
    margin: 2%;
}
```

```
#surface {
    padding-left: 0;
    padding-right: 0;
    margin-top: 5%;
    margin-left: auto;
    margin-right: auto;
    display: block;
    width: 50%;
    height: 20%;
}
```

```
#static/css/worker
```

```
#templates/index
```

```
<html>
```

```
<head>
```

```
    <meta http-equiv="refresh" content="5" >
```

```

<style>
  button {
    margin: 1%;
  }
img {
  width: 20%;
  margin-left: 50%;
  margin-bottom: 4%;
  margin-right: 50%;
  align-content: center;
}
</style>
</head>

<body>

  <button style="margin-bottom: 5%; display: block;"
onclick="window.location.href = 'all'">View all staff</button>

  {% for i in workers %}
    <button onclick="window.location.href = {{ i['id'] }}">{{ i['name'] }}</button>
  {% endfor %}
</body>
</html>
#templates/index
#templates/worker
<!DOCTYPE html>

```

```

<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>{{ name }}</title>
  <meta http-equiv="refresh" content="5" >
  <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}"
rel="stylesheet">
  <link href="{{ url_for('static', filename='css/worker.css') }}" rel="stylesheet">
</head>
<body>
  <div>
    <h2>{{ name }}!</h2>

    <table>
      <tr>
        <th>Works from time</th>
        <th>Is on work</th>
      </tr>
      <tr>
        <td>{{ start_time }}</td>
        <td>{{ working }}</td>
      </tr>
    </table>
  </div>
</body>
</html>
#templates/worker

```

```

#templates/worker_table
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Workers</title>
  <meta http-equiv="refresh" content="5" >
  <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}"
rel="stylesheet">
  <link href="{{ url_for('static', filename='css/worker.css') }}" rel="stylesheet">
  <script src="static/javascript/plotty.js"></script>
</head>
<body>
  <script>
    window.onload = function() {
      var some_y = {{ energy }};
      var some_x = [];

      for(i = 0; i < some_y.length; i++) {
        some_x.push(i + 1);
      }

      TESTER = document.getElementById('surface');

      var trace = {
        x: some_x,
        y: some_y,

```

```

    mode: 'lines+markers',
    name: 'Energy Consumption'
  };

```

```

var layout = {
  title: 'PµP±P»PµPSPëPµ CKP»PµPeC,CTPsCKPSPµCTPiPëPë'
};

```

```

var data = [ trace ];

```

```

Plotly.newPlot(TESTER, data, layout);

```

```

}

```

```

</script>

```

```

{% for i in workers %}

```

```

<div>

```

```

  <h2>{{ i['name'] }}!</h2>

```

```

  <table>

```

```

    <tr>

```

```

      <th>Last checked</th>

```

```

      <th>Is on work</th>

```

```

    </tr>

```

```

    <tr>

```

```

      <td>{{ i['start_time'] }}</td>

```

```

      <td>{{ i['working'] }}</td>

```

```

    </tr>

```

```

        </table>
    </div>
{% endfor %}

<div>
    <p> PтPёPSPёPjP°P»CHПSC<PNё PIPsP·PrCfC...PsPsP±PjPµPS: {{ value }}
</p>
    <div id="surface"></div>
</div>
</body>
</html>
#templates/worker_table
#main.config
[APP]
host=192.168.0.1
port=8888
consumption=60
#main.config
#requirements
flask
flask_cors
faker
#requirements

```