

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**  
**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ**  
**імені ІГОРЯ СІКОРСЬКОГО»**  
Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

До захисту допущено:

Завідувач кафедри

\_\_\_\_\_ Олександр Коваль

«\_\_» \_\_\_\_\_ 2020 р.

**Дипломна робота**

**на здобуття ступеня бакалавра**

**за освітньо-професійною програмою «Інформаційні технології моніторингу  
довкілля»**

**спеціальності 122 «Комп'ютерні науки та інформаційні технології»**

**на тему: «Веб додаток для оцінки ризику здоров'я населення від  
забруднення атмосферного повітря»**

Виконав (-ла):

студент (-ка) IV курсу, групи ТМ-62

Дібров Олег Юрійович \_\_\_\_\_

Керівник:

Ст-викл.

Гурін Артем Леонідович \_\_\_\_\_

Рецензент:

\_\_\_\_\_

Засвідчую, що у цій дипломній роботі  
немає запозичень з праць інших авторів  
без відповідних посилань.

Студент (-ка) \_\_\_\_\_

Київ – 2020 року

**Національний технічний університет України**  
**“Київський політехнічний інститут імені Ігоря Сікорського”**

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти перший рівень

Напрямок підготовки 122 Комп'ютерні науки та інформаційні технології

Спеціалізація Інформаційні технології моніторингу довкілля

**ЗАТВЕРДЖУЮ**

Завідувач кафедри

Олександр Коваль

(підпис)

” ” \_\_\_\_\_ 2020р.

**ЗАВДАННЯ**

**на дипломну роботу студенту**

Діброву Олегу Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи «Веб додаток для оцінки ризику здоров'я населення від забруднення атмосферного повітря»

керівник роботи \_\_\_\_\_ ст-викл. Гурін Артем Леонідович

(прізвище, ім'я, по батькові науковий ступінь, вчене звання)

затверджена наказом вищого навчального закладу від ”25” травня 2020р. № **1168-с**

2. Строк подання студентом роботи \_\_\_\_\_

3. Вихідні дані до роботи \_\_\_\_\_

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) проаналізувати існуючу систему оцінки ризику здоров'я населення від забруднення атмосферного повітря, спроектувати архітектуру веб-додатку для оцінки ризику здоров'я населення від забруднення атмосферного повітря, розробити програмне забезпечення, розробити інтерфейс користувача

5. Перелік ілюстративного матеріалу

1. Задача створення веб-додатку з оцінки ризику здоров'я населення від забруднення атмосферного повітря 2. Аналіз існуючих додатків 3. Опис предметної області 4. Засоби реалізації програмної системи 5. Опис програмної реалізації 6. Робота користувача з системою

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання ” ” \_\_\_\_\_ 20\_\_ р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітки
1.	Затвердження теми роботи		
2.	Вивчення та аналіз задачі		
3.	Розробка архітектури та загальної структури системи		
4.	Розробка структур окремих підсистем		
5.	Програмна реалізація системи		
6.	Оформлення пояснювальної записки		
7.	Захист програмного продукту		
8.	Передзахист		
9.	Захист		

Студент \_\_\_\_\_ Дібров О.Ю.

(підпис)

(прізвище та ініціали.)

Керівник роботи \_\_\_\_\_

Гурін А. Л.

(підпис)

(прізвище та ініціали.)

# ЗМІСТ

ВСТУП .....	4
1 Дослідження ризику здоров'я населення від забруднення атмосферного повітря .....	5
2 Аналіз існуючих додатків оцінки здоров'я населення .....	7
3 Опис предметної області .....	8
3.1 Забруднюючі речовини та їх класифікація.....	8
3.2 Методичні рекомендації щодо оцінки ризику для здоров'я населення від забруднення атмосферного повітря в Україні.....	11
3.2.1 Етапи оцінки ризику здоров'я .....	11
3.2.2 Алгоритм оцінки неканцерогенного ризику .....	16
3.2.3 Алгоритм оцінки канцерогенних ризику.....	18
4 Засоби реалізації програмної системи.....	20
3.1 Мова програмування Java .....	20
3.2 Фреймворк Spring .....	23
3.3 Фреймворк Spring Boot.....	24
3.4 Опис бази даних MySQL .....	25
3.5 Середовище розробки IntelliJ IDEA .....	26
3.6 Шаблон проектування MVC .....	27
3.7 Docker.....	28
5 Опис програмної реалізації .....	30
5.1 Проектування архітектури додатку.....	30
5.2 Use-case діаграма .....	32
5.3 Концептуальна модель бази даних.....	33
6 Робота користувача з системою.....	35
7 Висновки .....	41
8 Список використаних джерел .....	42

# ВСТУП

В екологічно забруднених регіонах з важкою промисловістю одним з основних чинників ризику для здоров'я населення є забруднення атмосферного повітря шкідливими речовинами.

В останні роки у світовій спільноті приділяється все більша увага розробці, вдосконаленню та впровадженню методології оцінки ризику для здоров'я населення під впливом факторів навколишнього середовища. Міжнародні організації — ВООЗ і ЄС широко використовують методологію оцінки ризику для характеристики якості атмосферного повітря, визначення збитків здоров'ю та оцінки економічної ефективності заходів щодо зниження викидів.

У той же час, буде актуально здійснення повсюдно фахівцями державної санітарно-епідеміологічної служби (СЕС) оцінки ризику для здоров'я населення в результаті забруднення атмосфери.

Метою роботи була створення веб додатку для оцінки ризику здоров'я населення від забруднення атмосферного повітря.

Перевищення концентрації хімічних речовин у повітрі на жаль є дуже частим явищем, тому продукт є актуальним. Записка містить 5 розділів.

- першому розділі описується постановка задачі з дослідження ризику здоров'я населення від забруднення атмосферного повітря
- другому аналіз існуючих додатків оцінки ризику здоров'я
- третьому розділі описується забруднюючі речовини та методичні рекомендації щодо оцінки ризику.
- четвертому розділі описані засоби веб додатку для оцінки ризику здоров'я.
- п'ятому розділі описані деталі реалізації програмного продукту.
- шостому розділі описано роботу користувача з системою.

# **1. ДОСЛІДЖЕННЯ ОЦІНКИ РИЗИКУ ЗДОРОВ'Я НАСЕЛЕННЯ ВІД ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ**

## **1.1 Створення веб додатку оцінки ризику здоров'я від забруднення атмосферного повітря**

У сучасному світі в екології необхідно завжди оцінювати ризики для здоров'я людей, а також світу в цілому. Тому задача оцінки буде відтворена по всіх методологій та рекомендацій. Метою даної дипломної роботи є створення веб додатку оцінки ризику здоров'я населення від забруднення атмосферного повітря, а саме інформування користувача про негативний вплив речовини на здоров'я людини при надлишковій концентрації.

При розробленні відповідного програмного забезпечення розв'язати потрібно наступні завдання:

1. Заповнити данні з речовинами до БД
2. Створення програмного коду для розрахування ризику в залежності від концентрації або дози отруйної речовини.
4. Обробити вхідні данні та вивести їх користувачу
5. Розробити вікно довідки для роз'яснення результату аналізу
6. Розробити вікно з мапою та вивести кількість викидів в атмосферне повітря за останні роки

## **1.2 Постановка задачі пошуку даних забруднюючих речовин**

У даній роботі використовуються офіційні дані бази даних нормативних документів Міністерства охорони здоров'я України.

## **1.3 Потенційні користувачі**

Система, що була розроблена може бути використана спеціалістами на будь-якій території (на підприємстві, техногенних аварій та ін.) з підвищеною концентрацією забруднюючої речовини і для оцінки ризику для здоров'я населення від атмосферного повітря. Для звітності по екології та медицини. Завдяки додатку можна зробити висновки які хімічні речовини більше всього впливають на життя людини. Зручний веб-інтерфейс, який доступний з будь-якого пристрою, надає можливість здійснювати аналіз даних.

## 2. АНАЛІЗ ІСНУЮЧИХ ДОДАТКІВ ОЦІНКИ ЗДОРОВ'Я НАСЕЛЕННЯ

У наш час систем, аналогічних до розробленої, немає. Але, схожий результат можна отримати з ресурсу ВОЗ, а саме програма AirQ+. Інструмент AirQ + був створений для того, щоб полегшити розрахунок того збитку, який забруднення повітря завдає здоров'ю окремих груп населення. Зокрема, інструмент AirQ + допомагає знайти відповідь на такі важливі питання:

1) В якій мірі досліджуваний ефект на здоров'я пов'язаний із забрудненням повітря

2) Яким було б тягар хвороб, якби рівень забруднення повітря був не таким як у даний час (при цьому, як правило, передбачається, що він буде нижче)

3) Якою мірою зміняться ефекти на здоров'я в майбутньому в разі зміни рівнів забруднення повітря в порівнянні з поточним сценарієм? Всі розрахунки, виконані інструментом AirQ +, засновані на методології і функціях “концентрація-відповідь”, встановлених в ході епідеміологічних досліджень. Функції концентрація-відповідь, що використовуються інструментом AirQ +, засновані на систематичному огляді всіх доступних досліджень (до 2013 року) і їх метааналізу.

Недоліком цієї програми є відсутність бази даних отруйних речовин, а також неможливість отримання результату канцерогенного ризику здоров'я населення від забруднення атмосферного повітря.

Тому необхідна система для розробки – це система, легка в користуванні та з можливістю запуску на будь-якій системі. Також система повинна бути з заповненою БД з основними забруднюючими речовинами.

### **3. ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ**

#### **3.1 Забруднювачі речовини та їх класифікація**

Забруднювачі речовини – це речовини, які негативно впливають на навколишнє середовище, безпосередньо чи після хімічних змін в атмосфері або в поєднанні з іншими речовинами та забруднювачами.

Атмосферне повітря є основним середовищем існування біосфери, включаючи людину. В результаті розвитку цивілізації розвинулася еволюція Землі, і постійне співвідношення основних компонентів повітря істотно не змінилося.

Основні гази в атмосфері (азот, кисень, аргон) по суті прозорі і розсіюють довгохвильове та короткохвильове випромінювання. На екологію значно впливають газоподібні забруднювачі, які можуть бути природними та техногенними. Їх природні джерела включають вітрову ерозію, вивільнення солі з океанів і морів, вулканічні та біологічні процеси та викиди з космосу.

Теплові електростанції - одне з основних джерел забруднення навколишнього середовища. Останні дані експертів із охорони здоров'я свідчать, що токсичний вплив хімічних речовин у поєднанні зі шумом та вібрацією зростає в 2,5-3 рази. В результаті хімічної взаємодії двох токсичних речовин можуть бути синтезовані нові шкідливі для людини компоненти. При взаємодії канцерогенних вуглеводнів та оксидів азоту синтезуються сполуки, що синтезують людський генофонд.

Джерелами забруднення повітря є природні та антропогенні процеси (джерела, що виникають у житті людини). Обсяг природних викидів шкідливих речовин може бути в кілька разів більшим, ніж ті, які виробляє людина. Однак природні викиди розподіляються рівномірно по всій атмосфері і створюють лише фонові концентрації. У той же час антропогенні викиди відбуваються постійно і мають локальний характер, спричиняючи небезпечні концентрації шкідливих речовин у природному середовищі. Викиди забруднюючих речовин

повітря можуть бути високими або відносно низькими. Залежно від цього домішки поділяються на масові та питомі. До масових забруднювачів належать: сірчистий газ, оксиди азоту, чадний газ, вуглеводні та пил. Газоподібні речовини та пари, схожі на пари, що виділяються в атмосферу у відносно невеликій кількості, називаються забруднювачами повітря. До них належать аміак, бензо (а) пірен, сірководень, сірковуглець, галогени (хлор, фтор) та їх сполуки, пари ртуті, меркаптани, діоксини та інші. Ці сполуки зазвичай знаходяться у повітрі поблизу промислових заводів, які або використовують ці речовини в технологічних процесах, або утворюються при виробництві інших хімічних сполук.

Викиди від транспортних засобів - вихлопні гази, що виділяються в навколишнє середовище під час експлуатації транспортних засобів. Транспортні засоби широко застосовуються в будівництві та будівельній галузі. Склад викидів автомобіля залежить головним чином від виду палива. В даний час основним видом палива є бензин, меншою мірою дизель, скраплений природний газ та інші. Загальна кількість речовин, що містяться у викидах автомобілів, перевищує 1000 статей.

Найнебезпечнішими для здоров'я людини є вуглеводні, особливо поліциклічні ароматичні речовини, оксид вуглецю, азот і сірка, тверді речовини. Основними причинами збільшення вмісту токсичних речовин у викидах автомобіля є неповне згорання палива через неправильне регулювання карбюратора та поганий технічний стан двигуна. Викиди від транспортного засобу включають також кривошипи - суміш парів газу та масла. Випаровування летких фракцій бензину з паливних баків, карбюраторів, паливних систем двигуна також може бути джерелом забруднення повітря.

Автомобільні двигуни виділяють більше 95% оксиду вуглецю, близько 65% вуглеводнів та 30% оксидів азоту у повітря міста. При спалюванні 1 кг бензину в атмосферу надходить 465 г окису вуглецю, 25 г вуглеводнів, 15 г оксидів азоту. Крім того, для спалювання 1 кг бензину необхідно 14,5 кг повітря. Це означає, що двигун внутрішнього згорання (ДВЗ)

Споживання близько 200 літрів кисню на годину - в 2,5 рази більше, ніж одна людина вдихає в день. Частка двигунів з іскровим запалюванням в загальному забрудненні повітря токсичними викидами автомобілів становить 93-95%, а дизельних двигунів 5-7%.

Шкідливий вплив викидів автотранспорту на людину виражається в збільшенні гострих і хронічних захворювань легенів і серцево-судинної системи, а також раку.

Електрозварювання - це метод зварювання металів шляхом нагрівання і плавки за допомогою електрики. Електричний нагрів дозволяє отримувати температури, що перевищують температуру плавлення металу, без зміни його хімічного складу. Електрозварювання - найважливіший вид зварювання металу в будівництві, який займає перше місце за обсягом робочого навантаження, кількості існуючих об'єктів і кількістю працівників. Процес електрозварювання характеризується параметрами зварювальної дуги, структурою зварювання і технологічними параметрами виробництва.

Кількість забруднень, що виділяються при зварюванні або зливі поверхонь, характеризується загальним викидом в 1 кг витратних матеріалів для зварювання. У процесі різання металу певні значення виражаються в грамах на метр довжини різання і мають різні значення в залежності від товщини металу, що розрізає.

Докладні показники забруднення при виробництві різних зварювальних і поверхневих робіт представлені у вигляді таблиць в методиці розрахунку викидів (викидів) забруднюючих речовин в атмосферу при зварюванні. Питомі показники виділення забруднюючих речовин при виробництві різних зварювально-наплавочних робіт представлені у вигляді таблиць в методиці розрахунку виділень (викидів) забруднюючих речовин в атмосферу при зварювальних роботах.

## 3.2 Методичні рекомендації щодо оцінки ризику для здоров'я населення від забруднення атмосферного повітря в Україні

### 3.2.1 Етапи оцінки ризику здоров'я

Методологія оцінки ризику – це вибір оптимальних у даній конкретній ситуації шляхів усунення або зменшення ризику, він складається з трьох взаємопов'язаних блоків (Таблиця 3.1)

- 1) оцінка ризику;
- 2) управління ризиком;
- 3) інформування про ризик.

Назва блоку	Характеристика шляхів усунення або зменшення ризику
Оцінка ризику	Повна, або базова, схема оцінки ризику передбачає проведення чотирьох взаємопов'язаних етапів, а саме:  1) ідентифікацію небезпеки;  2) оцінку експозиції;  3) характеристику небезпеки (оцінку залежності “доза-відповідь”);  4) характеристику ризику.
Управління ризиком	Основні завдання управління ризиком – порівняльне вивчення факторів ризику, установлення вагомості ризиків, їхнє ранжування і виявлення пріоритетів, обґрунтування найкращих в даній ситуації рішень з усунення або мінімізації ризику, а також оцінка ефективності і корегування оздоровчих заходів. Управління ризиком базується на сукупності політичних, соціальних і економічних оцінок отриманих величин

	ризиків, порівняльній характеристиці можливої шкоди для здоров'я людини і суспільства в цілому, можливих витрат на реалізацію різних варіантів управлінських рішень зі зниження ризику і тих вигод, які будуть отримані в результаті реалізації заходів.
Інформування про ризик	Інформування про ризик – це процес розповсюдження результатів визначення ступеня ризику для здоров'я людини і рішень щодо його контролю. На їх основі органи санепідслужби спільно з адміністративними органами, з огляду на пріоритетність як окремих джерел забруднення, так і провідних чинників, які формують найбільш високий і небезпечний рівень ризику для здоров'я населення та стану навколишнього природного середовища, розробляють комплекс профілактичних заходів і черговість їх впровадження. Цей аспект є принципово новим і відрізняє концепцію ризику від попередніх концепцій, що використовувались при оцінці небезпеки впливу шкідливих факторів оточуючого середовища на населення.

Таблиця 3.1 — Характеристика основних блоків методології оцінки ризику

Основним завданням першого етапу (ідентифікація небезпеки) є вибір першочергових та індикаторних хімікатів, тестування яких дозволить забезпечити достатню характеристику рівня ризику порушень здоров'я та їх джерел. Пріоритетність випробуваних речовин визначається на основі даних про їх біологічну активність, включаючи канцерогенні, фізико-хімічні властивості, що визначають особливості їх зовнішнього вигляду та поведінки в природному середовищі та вплив на організм людини, залежність від розвитку негативних впливів (специфічних та неспецифічних) речовини для організму. У цьому випадку зазвичай слід використовувати вторинні джерела інформації (аналітичні огляди, звіти, каталоги, бази даних), які вже містять думку висококваліфікованих експертів щодо небезпечних властивостей цієї речовини.

Оцінка експозиції - це другий етап оцінки ризику, на якому певним чином визначається кількісний рівень введення речовини в організм людини. Сюди входить визначення маршруту поширення в навколишньому середовищі та вплив забруднювача на організм, випробування його концентрацій, визначення тривалості опромінення, оцінка чисельності популяції, на яку впливають шкідливі фактори чи можуть бути вплинуті ними.

Кількісні характеристики впливу засновані на визначенні концентрації хімічних сполук, які впливають на людей, виходячи з таких даних:

- моніторингові тести;
- моделювання поширеності та поведінки хімічних сполук у повітрі (природне середовище);
- поєднання результатів моніторингу спостереження з даними моделі.

Моніторинг якості атмосферного повітря є найважливішим інструментом для аналітичного визначення вмісту хімічної речовини. У сучасних умовах джерелом даних можуть бути результати спеціально спрямованих спостережень та матеріалів щодо забруднення повітря, одержані системою державного моніторингу Державної гідрометеорологічної служби Міністерства з надзвичайних ситуацій України та Державної санітарно-епідеміологічної служби Міністерства охорони здоров'я України.

Концентрація речовини в зоні спостереження (місце розташування людини) визначається як середнє арифметичне концентрацій, що відбулися протягом періоду опромінення, або як максимальна концентрація протягом обмеженого часу (залежно від завдання). Для оцінки ризику хронічного впливу хімічних речовин слід використовувати щорічні концентрації ризику та їх верхній 95% довірчий інтервал. При визначенні ризику виникнення кризи (екстремальних, надзвичайних ситуацій) до 24 годин. застосовуються максимальні концентрації. Для України важливо зосередитись на списках загальних забруднювачів повітря [1].

Результатом цього етапу оцінки ризику є визначення середньодобової дози (ADD / LADD), формулу якої для обчислення за інгаляційного впливу з



будь-яких доз, що викликають пошкодження генетичного матеріалу; для такого роду сполук *відсутні порогові рівні*. [2]

Для характеристики ризику неканцерогенних ефектів найчастіше використовують два показники:

- 1) максимальна ефективна доза;
- 2) мінімальна доза, яка визначає пороговий ефект.

Ці показники є основою для визначення мінімальних рівнів дози ризику - еталонних (RfD) та концентрації (RfC). Перевищення еталонної дози не обов'язково пов'язане з розвитком побічних ефектів, але чим вище і вище доза перевищує еталонну, тим більша ймовірність її виникнення, але неможливо оцінити цю ймовірність за допомогою цього методологічного підходу. У цьому відношенні кінцевими характеристиками оцінки впливу на основі референтних доз та концентрацій є фактори (HQ) та показники небезпеки (HI). Якщо референтна доза не буде перевищена, не потрібно застосовувати регуляторне втручання. Коли речовина перевищує RfD, виникає небезпека, яку можна оцінити лише досліджуючи залежність доза-відповідь та спектр несприятливих ефектів.

Для оцінки ризику генотоксичних канцерогенів основним параметром є потенційний канцероген (КПК) або фактор нахилу (СФ), який відображає підвищений ризик канцерогенного ризику на одиницю із збільшенням дози та має розмір (мг / кг x добу) <sup>-1</sup>.

Ще один параметр - значення т. Зв Одиничний ризик (RU). Під впливом інгаляційного впливу УР є надзвичайно консервативною оцінкою канцерогенного ризику для людини, який піддається постійній тривалості життя конкретного канцерогену в концентрації 1 мкг / м<sup>3</sup>.

Четвертий етап – характеристика ризику інтегрує дані про небезпеку досліджуваних речовин, величину експозиції, параметри залежності “доза-відповідь”, які було отримано на попередніх етапах дослідження. На основі цих даних дається кількісна та якісна оцінка ризику окремих речовин та визначається порівняльний ряд небезпеки для здоров'я населення групи сполук. [3]

### 3.2.2 Алгоритм розрахунку неканцерогенного ризику

Характеристика ризику для неканцерогенних ефектів проводиться шляхом порівняння фактичних рівнів впливу з безпечним (референтним) рівнем впливу та визначення небезпеки:

$$HQ = \frac{AD}{RfD}, \quad \text{або} \quad HQ = \frac{AC}{RfC}, \quad (3.1)$$

де  $HQ$  – коефіцієнт небезпеки;

$AD$  – середня доза, мг/кг;

$AC$  – середня концентрація, мг/м<sup>3</sup>;

$RfD$  – референтна (безпечна) доза, мг/кг;

$RfC$  – референтна концентрація, мг/м<sup>3</sup>.

На думку зарубіжних експертів, за відсутності еталонних доз - концентрацій гранично допустимі концентрації (ГДК) або граничні рівні або концентрації бездіяльності (МНР, МНК), визначені за критерієм прямого впливу на здоров'я, можуть використовуватися як еквівалент.

Для інгаляції, якщо для цього не потрібні спеціальні дослідницькі завдання, не потрібно розраховувати дозу впливу, а розрахунок коефіцієнта ризику можна проводити за формулою:

$$HQ_i = \frac{C_i}{RfC}, \quad (3.2)$$

де  $HQ_i$  – коефіцієнт небезпеки впливу  $i$ -тої речовини;

$C$  – рівень впливу  $i$ -тої речовини, мг/м<sup>3</sup>;

$RfC_i$  – безпечний рівень впливу, мг/м<sup>3</sup>.

Коефіцієнт небезпеки розраховують окремо за умов короткотривалого (гострого), підгострого і тривалого впливу хімічної речовини. При цьому період

осереднення експозиції і відповідних безпечних рівнів впливу має бути аналогічним. [4]

Характеристика ризику	Коефіцієнт небезпеки ( $HQ$ )
Ризик виникнення шкідливих ефектів розглядають як зневажливо малий	< 1
Гранична величина, що не потребує термінових заходів, однак не може розглядатися як досить прийнятна	1
Імовірність розвитку шкідливих ефектів зростає пропорційно збільшенню $HQ$	> 1

Таблиця 1 — Критерії неканцерогенного ризику

Характеристика ризику для неканцерогенних ефектів із комбінованою дією хімічних речовин ґрунтується на обчисленні коефіцієнта небезпеки за формулою:

$$HI = \sum HQ_i, \quad (3.3)$$

де  $HQ_i$  – коефіцієнти небезпеки для окремих забруднюючих речовин.

Розрахунок показників небезпеки зазвичай проводиться з урахуванням ключових органів і систем, на які негативно впливають речовини, які випробовуються. За результатами наукових досліджень під впливом компонентів суміші на ті самі органи або системи організму найбільш ймовірним видом комбінованого впливу є добавка (адиктивність). Це правило не є універсальним, оскільки не враховує можливих відмінностей у конкретних механізмах дії компонентів суміші, а також місцевих побічних ефектів у первинному місці контакту з організмом (наприклад, слизових оболонок дихальних шляхів або слизової шлунка).

Однак, на думку міжнародних та зарубіжних експертів, такий підхід може перебільшувати ризики для здоров'я, але має більшу перевагу перед окремою та незалежною оцінкою кожного елемента. [5]

### 3.2.3 Алгоритм розрахунку канцерогенного ризику

Для характеристики канцерогенного ризику проводяться розрахунки індивідуального ризику та популяції, що піддаються дії досліджуваних речовин.

Розрахунок індивідуального канцерогенного ризику проводиться за формулою:

$$CR = SF \cdot LADD, \quad (3.4)$$

де  $CR$  – імовірність занедужати раком, безвимірна величина (звичайно виражається в одиницях 1:1000000);

$SF$  – фактор нахилу, тобто імовірність одержання ракового захворювання у випадку прийому одиничної дози  $LADD$ ,  $1/\text{мг}/(\text{кг} \times \text{доба})$ ,  $((\text{мг}/(\text{кг} \times \text{доба}))^{-1})$ ;

$LADD$  – середня добова доза протягом життя,  $\text{мг}/(\text{кг} \times \text{доба})$ .

При застосуванні величини одиничного ризику розрахункова формула набуває вигляду:

$$CR = LADC \cdot UR, \quad (3.5)$$

де  $LADC$  – середня концентрація речовини в атмосферному повітрі за весь період усереднення експозиції,  $\text{мг}/\text{м}^3$ ;

$UR$  – одиничний ризик,  $(\text{мг}/\text{м}^3)^{-1}$ .

Одиничний ризик ( $UR$ ) розраховують із використанням величини  $SF$ , стандартної величини маси тіла людини (70 кг) та добового споживання повітря ( $20 \text{ м}^3$ ):

$$UR_i (\text{м}^3/\text{мг}) = SF_i (\text{мг}/\text{кг} \times \text{доба})^{-1} \times 1/70 \text{ кг} \times 20 (\text{м}^3/\text{доба}). \quad (3.6)$$

Поряд з розрахунками індивідуального канцерогенного ризику проводять визначення популяційного ризику ( $PCR$ ), який відображає додаткову (до фонові) кількість випадків новоутворень, які можуть виникнути протягом життя внаслідок впливу досліджуваного фактора:

$$PCR = CR \cdot POP, \quad (3.7)$$

де  $CR$  – індивідуальний канцерогенний ризик;

$POP$  – чисельність популяції, що підпадає під вплив даного фактору, чол.

При порівняльній характеристиці ризику іноді використовують величину популяційного річного ризику ( $PCRa$ ), що являє собою розраховану кількість додаткових випадків раку протягом року:

$$PCRa = \sum (C_i \cdot UR_i) \cdot POP / 70, \quad (3.8)$$

де  $C_i$  – середня річна концентрація  $i$ -тої речовини;

$POP$  – чисельність популяції, що зазнає впливу, чол.

$UR_i$  – одиничний ризик протягом життя (70 років).

Канцерогенний ризик, пов'язаний з комбінованим впливом кількох хімічних речовин, вважається добавкою. Аналіз рекомендує групувати досліджувані канцерогени з урахуванням типу та / або розташування пухлин. У цьому випадку розрахунок загального канцерогенного ризику проводиться окремо для кожної групи (наприклад, при раку легень, раку печінки тощо).

Таким чином, під впливом декількох канцерогенів загальний канцерогенний ризик розраховується за формулою:

$$CR_T = \sum CR_j, \quad (3.9)$$

де  $CR_T$  – загальний канцерогенний ризик для шляху надходження  $T$ ;

$CR_j$  – канцерогенний ризик для  $j$ -тої канцерогенної речовини.

## 4. ЗАСОБИ РЕАЛІЗАЦІЇ ПРОГРАМНОЇ СИСТЕМИ

Під час розробки веб додатку можна виділити такі основні етапи:

1. Створення БД для збереження інформації
2. Розробка серверної частини (backend). При розробці додатку була використана мова програмування Java
3. Розробка UI (frontend). При розробці була використана мова програмування JavaScript з HTML та CSS

### 4.1 Мова програмування Java

Java — об'єктно-орієнтована мова програмування, розроблена Sun Microsystems з 1991 року та офіційно випущена 23 травня 1995 року. Спочатку нова мова програмування називалася Oak (Джеймс Гослінг) і була розроблена для побутової електроніки, але пізніше її перейменували в Java і почали використовувати для написання аплетів, програм та серверних програм.

Програми Java можуть бути переведені в байт-код, що працює на віртуальній машині Java (програма JVM, яка обробляє байт-код), яка передає інструкції до апаратного забезпечення як перекладача, але з тією різницею, що байт-код, на відміну від тексту, обробляється набагато швидше. Переваги цього способу виконання програм — у повній незалежності байт-коду від ОС та обладнання, що дозволяє запускати програми Java на будь-якому пристрої, що підтримує віртуальну машину. Ще однією важливою особливістю технології Java є гнучка система захисту, завдяки тому, що виконання програми повністю контролюється віртуальною машиною. Будь-які операції, що перевищують встановлені повноваження програми (наприклад, спроба отримати несанкціонований доступ до даних або підключитися до іншого комп'ютера), викликають миттєве переривання. Це дозволяє користувачам завантажувати програми, написані на Java, на свої комп'ютери (або інші пристрої, такі як

мобільні телефони) з невідомих джерел, не боячись зараження вірусом, втрати цінної інформації тощо. Часто до недоліків такого підходу можна віднести те, що виконання байт-коду віртуальною машиною може знизити продуктивність програм і алгоритмів, реалізованих на Java. Це твердження справедливо для перших версій віртуальної машини Java, але останнім часом воно практично втратило актуальність. Цьому сприяла низка вдосконалень: використання технології JIT (Just-In-Time Compiler), яка дозволяє переводити байт-код у машинний код під час виконання програми з можливістю збереження версій класу в машинному коді, широкого використання Native коду в стандартних бібліотеках та апаратних засобах, які забезпечує прискорену обробку байт-коду (наприклад, технологія Jazelle, що підтримується деякими процесорами ARM).

[6]

Мова програмування Java має наступні характеристики:

- Простота мови це одна з головних особливостей Java: розробнику не доведеться вивчати мову задовго до того, як він зможе програмувати на ній. Основні поняття мови Java швидко розуміються, і програмісти можуть продуктивно працювати з самого початку. Розробники Java врахували, що багато програмістів знайомі з мовою C ++, тому Java максимально наближений до C ++;
- об'єктно-орієнтовані. З самого початку мова Java розроблялася як об'єктно-орієнтована. Завдання розповсюдження клієнтської системи відповідає об'єктно-орієнтованій парадигмі: використання понять інкапсуляція, успадкування та поліморфізм. Java забезпечує чітку та ефективну об'єктно-орієнтовану платформу розвитку. Java-програми можуть використовувати стандартні бібліотеки об'єктів, які надають пристрої вводу / виводу, функції мережі та способи створення графічних інтерфейсів користувача;
- надійність. Платформа Java створена для створення високонадійного програмного забезпечення. Велика увага приділяється перевірці програми на етапі складання, далі йде другий рівень — динамічна перевірка (на етапі впровадження);
- Незалежність від архітектури. Java призначена для підтримки додатків, вбудованих у різні мережеві середовища. За таких умов програми

повинні працювати на різних апаратних архітектурах, запускати різні операційні системи та взаємодіяти з інтерфейсами на різних мовах програмування. Для забезпечення незалежності платформи компілятор Java генерує байт-код;

- архітектурно проміжний формат програми який розроблен для ефективної передачі коду на різних апаратних та програмних платформах. Коли програма виконується, байт-код інтерпретується комп'ютером, який працює з Java. Той самий Java byte code буде працювати на будь-якій платформі;

- Універсальне зчитування. Більшість сучасних мережевих програм зазвичай потребують виконання декількох дій одночасно. Java реалізує механізм підтримки легких процесів. Потік Java надає інструменти для створення додатків з декількома одночасно активними потоками;

- Висока продуктивність. Продуктивність завжди заслуговує на особливу увагу. Java досягає високої продуктивності завдяки спеціально оптимізованому байтовому коду, який можна легко перевести в машинний код. Автоматичне збирання сміття виконується як фоновий потік з низьким пріоритетом, що забезпечує високу ймовірність доступу до потрібної пам'яті, що призводить до підвищення продуктивності. Програми, які потребують великих обчислювальних ресурсів, можуть бути розроблені таким чином, що ті частини, які потребують інтенсивних обчислень, записуються під час створення та взаємодіють із платформою Java.

## 4.2 Фреймворк Spring

Платформа Spring - популярна платформа додатків із відкритим кодом, призначений для спрощення розробки в J2EE. Він складається з контейнера Платформи управління елементами та набір інтегрованих служб для інтерфейсів користувача, транзакцій та збереження статусу Веб-платформа Spring включає в себе MVC, розширювану платформу MVC для створення веб-додатків. (рисунок 3.1)

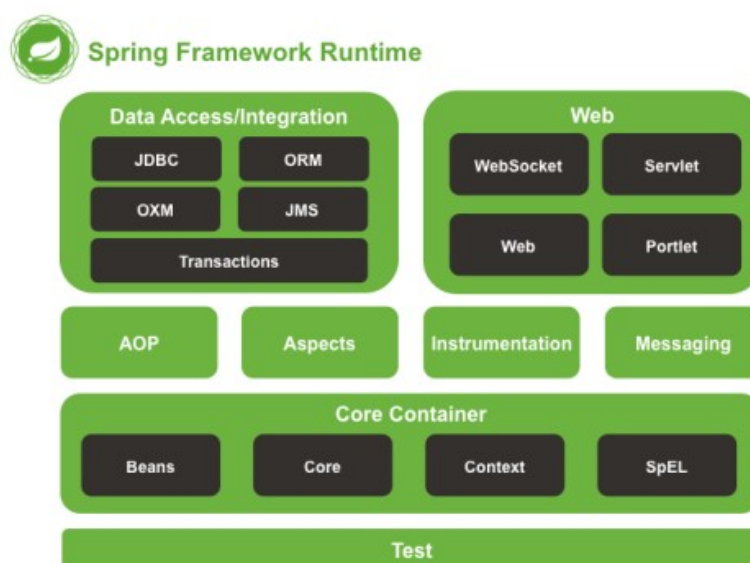


Рисунок 3.1. Структура фреймворка Spring

Хоча Spring не пропонував особливого об'єктного програмування, широко поширена в основній спільноті Java як альтернатива та заміна моделі Enterprise JavaBeans. Налаштування Spring дає програмістам Java велику свободу дизайну; додаткові пропозиції добре задокументовані та прості у використанні засоби для вирішення цієї проблеми виникають під час створення програм у корпоративному масштабі Spring Framework пропонує рішення багатьох завдань, за допомогою яких Java-програмісти та організації, які хочуть створювати інформацію система на базі платформи Java. Це важко через свою широку функціональність визначити найбільш значущі структурні елементи, з яких він складається. Spring Однак оточення не повністю пов'язане з платформою Java Enterprise широка інтеграція з ним, що є важливою причиною його популярності.

Spring Framework - це, мабуть, найвідоміше джерело розширень (функцій), необхідні для ефективного розвитку складних бізнес-застосунків важкі моделі програмного забезпечення, які історично домінували промисловість . Ще одна перевага полягає в тому, що вона була введена раніше функціональність, що не використовується в сучасних домінуючих методах програмування навіть поза платформою Java. Ця рамка забезпечує узгоджений зразок і робить його придатним більшість типів додатків, які вже побудовані на основних платформах Java. Вважається що Spring Framework реалізує модель розробки, засновану на найкращих стандартах і робить його доступним у багатьох областях Java. [7]

### 4.3 Фреймворк Spring Boot

Spring Boot - це програмне забезпечення з відкритим кодом на основі Java, яке використовується для створення мікросервісів. Він розроблений командою Pivotal і використовується для створення автономних і готових Spring додатків. [8]

#### Переваги

- Легкий у виконанні
- Проста масштабованість
- Сумісний з контейнерами
- Мінімальна конфігурація
- Легкий для розуміння та розвитку Spring додатків
- Підвищує продуктивність
- Скорочує час програмування

Spring Boot був розроблений для таких цілей

- Щоб уникнути складної конфігурації XML
- Простіший спосіб розробити програми
- Скоротити час програмування та самостійно запуснути програму

Забезпечує ефективну пакетну обробку та керує цілями REST. Все автоматично налаштовується у Spring Boot. Конфігурується Spring Boot у файлі application.properties. Містить вбудований контейнер сервлетів

### **3.3 . Опис бази даних MySQL**

На даний момент MySQL є однією з найпопулярніших систем управління базами даних в Інтернеті. Ця система використовується для роботи з досить великою кількістю інформації. Однак MySQL ідеально підходить як для невеликих проектів, так і для Інтернету. Важливою особливістю системи є її свобода.

Основні переваги MySQL це надійність, швидкість та гнучкість - основні особливості MySQL. Робота з цією системою не є складною, і підтримка сервера MySQL автоматично включається в доставку PHP. MySQL випускається за умовами Загальної публічної ліцензії GNU (GPL) [9].

Раніше файли використовувались для тривалого та безпечного зберігання інформації: в них було збережено кілька рядків, які потім були вилучені для подальшої роботи. Проблема довготривалого зберігання інформації є досить важливою в процесі програмування веб-додатків. Наприклад, якщо мова йде про підрахунок кількості відвідувачів сайту на касі, зберігання повідомлень на форумі, а якщо ви хочете віддалено керувати даними на сайті, не використовуючи систему для тривалого зберігання інформації, цього недостатньо.

Однак професійні методи роботи з файлами досить складні та трудомісткі, тому що ви повинні бути обережними для введення інформації у файли, сортування даних та цих тверджень. Однак пам'ятайте, що всі ці дії виконуватимуться на сервері хостинг-провайдера, на якому, ймовірно, встановлено один із параметрів Unix. Тому ви також повинні забезпечити безпеку доступу до файлів. У цьому випадку кількість коду значно збільшується і є велика ймовірність помилки в програмі.

Вищезазначені завдання були успішно вирішені за допомогою бази даних, яка координує захист інформації, сортує її та дозволяє витягувати та розміщувати дані в одному рядку. Код, що використовує базу даних, здається більш компактним, що полегшує вирішення. Крім того, ми не можемо забути показники швидкості: відбір інформації з бази даних відбувається швидше, ніж з файлів.

### **3.4 Середовище розробки IntelliJ IDEA**

Перша версія IntelliJ IDEA з'явилася в січні 2001 року і швидко стала популярною як перша Java IDE з широким спектром вбудованих інструментів рефакторингу, що дозволило програмістам швидко реорганізувати вихідний код програми.

Дизайнерське середовище фокусується на продуктивності програми, що дозволяє зосередитись на розробці функціональних можливостей, в той час як IntelliJ IDEA бере на себе звичайні операції.

Починаючи з шостої версії продукту, IntelliJ IDEA пропонує інтегровані інструменти для створення графічного інтерфейсу користувача. Серед інших особливостей, IntelliJ IDEA добре сумісний з багатьма популярними програмами програмістів з відкритим кодом, такими як CVS, Subversion, Apache Ant, Maven та JUnit. Починаючи з версії 9.0, IntelliJ IDEA доступний у двох версіях: Community Edition та Ultimate Edition.

Глобальна версія є повністю відкритим кодом, доступний за ліцензією Apache 2.0. Він повністю підтримує Java SE, Groovy, Scala та інтеграцію з найпопулярнішими системами управління версіями. Ultimate Edition підтримує діаграми Java EE, UML, розрахунки покриття коду та підтримку інших версій, систем управління мовою та кадрами (рисунок 3.2).

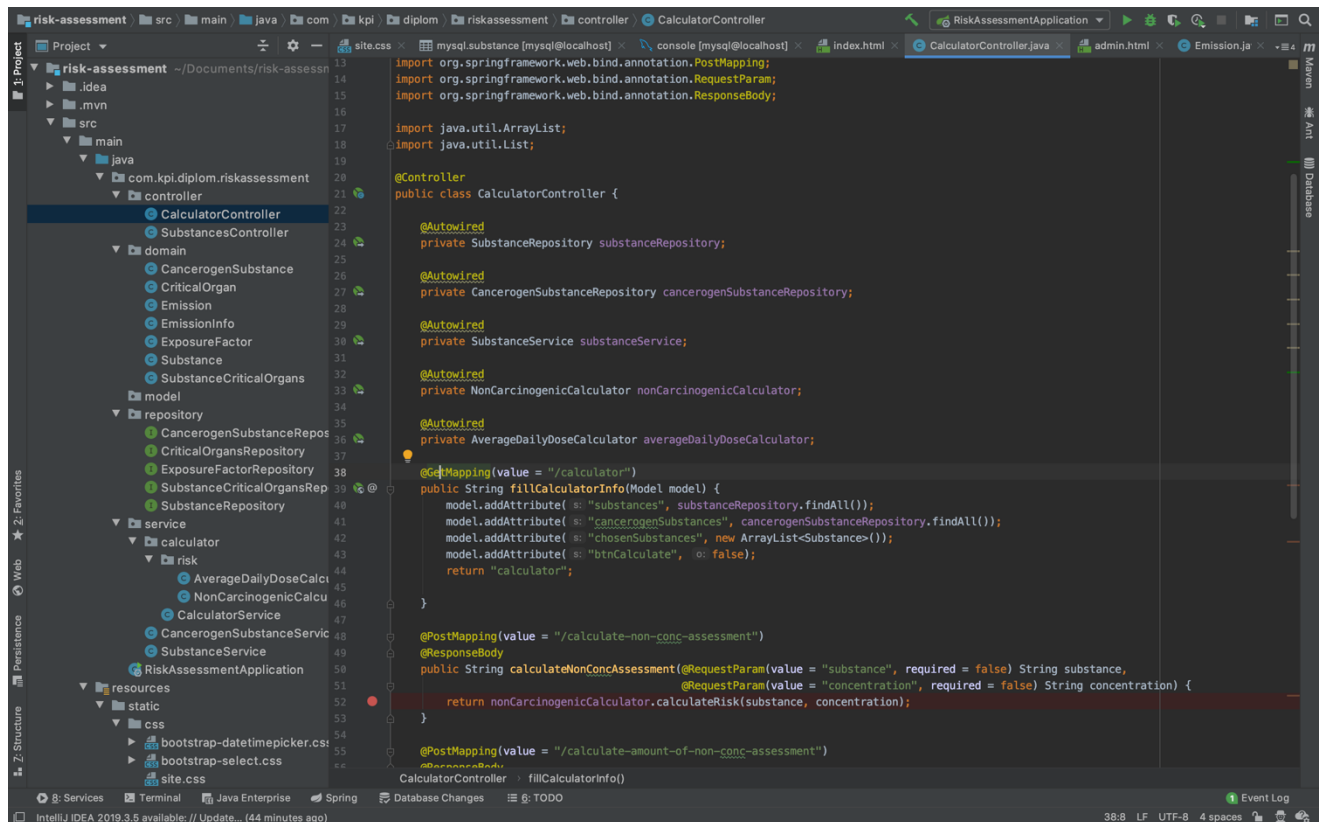


Рисунок 3.2 — Середовище розробки JetBrains IntelliJ Idea 2019.3

## 3.5 Шаблон проектування MVC

Принцип MVC в веб-програмуванні (Модель — Вид — Контролер, Модель — Вид (Вид) — Контролер) — одна з найуспішніших ідей сьогодні. Принцип MVC на перший погляд інтуїтивно зрозумілий, але глибина не дуже проста. Для початку розглянемо, для чого він призначений (Рисунок 3.3) .

MVC допомагає розділити реалізацію логіки докладання, зовнішнього вигляду (графічний інтерфейс) і взаємодії з користувачем. Це призводить до більш структурованого та добре написаного коду, дозволяє більш спеціалізованим людям працювати над проектом, спрощує підтримку коду, робить його більш логічним і зрозумілим. Зміна одного компонента мінімально впливає на інші. Різні типи різних контролерів можуть бути підключені до однієї моделі. Розглянемо компоненти більш докладно. Модель — містить так звану “бізнес-логіку” — обробка та перевірка даних, доступ до баз даних, являє собою

внутрішню структуру системи. Модель не повинна взаємодіяти безпосередньо з користувачем. Вид описує зовнішній вигляд програми. Контролер це зв'язок між моделлю і представленням, приймає дані від користувача, передає їх в модель, отримує оброблений результат і передає його для перегляду.

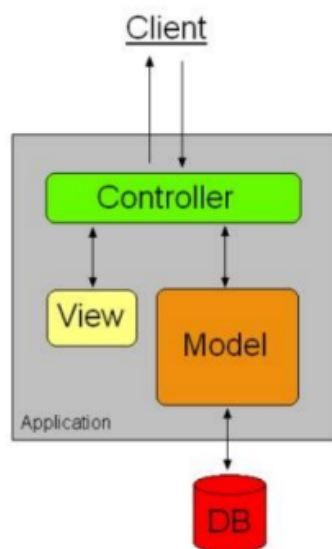


Рисунок 3.3 Модель MVC

Spring Framework надає можливість для реалізації архітектури MVC через Spring WEB MVC framework.

### 3.5 Docker

Docker - це платформа для програмістів та системних адміністраторів для створення, запуску та обміну контейнерними програмами. Використання контейнерів для розгортання програм називається контейнеризацією. Контейнери не нові, але їх можна легко реалізувати.

## Переваги контейнеризації:

- Гнучкість: навіть найскладніші програми можна помістити в контейнер.
- Простота: контейнери використовують і діють ядро хоста, що робить їх набагато ефективнішими ресурсами, ніж віртуальні машини.
- Портативний: ви можете створювати локально, реалізовуватися у хмарі та працювати в будь-якому місці.
- Низька прив'язка: контейнери дуже незалежні і укладені в корпус, що дозволяє їх замінювати або оновлювати, не заважаючи іншим.
- Масштабованість: Ви можете автоматично збільшувати та розповсюджувати репліки контейнерів через центр обробки даних.
- Безпека: контейнери застосовують агресивні обмеження та ізоляції до процесів без необхідної конфігурації користувача.

В основному контейнер - це не що інше, як процес згортання, який використовує додаткові функції інкапсуляції, тому він ізольований від хоста та інших контейнерів. Одним з найважливіших аспектів ізоляції контейнерів є те, що кожен контейнер працює з власною приватною файловою системою; ця файлова система має зображення Докера. Зображення містить все необхідне для запуску програми - двійковий код або файл, середовище виконання, залежності та будь-які інші об'єкти файлової системи [10].

## 4. ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

Реалізація веб додатку для оцінки ризику здоров'я населення від забруднення атмосферного повітря складається з проектування архітектури додатку, створення бази даних, реалізацію серверної частини, створення веб частини, та тестування.

### 4.1 Проектування архітектури додатку

Веб додаток був розроблен з дотриманням архітектури MVC (Model, View, Controller).

- Модель (Model) - це дані і правила, які використовуються для роботи з даними, які представляють концепцію управління додатком. У будь-якому додатку вся структура моделюється як дані, які обробляються певним чином. Що таке користувач для додатка - повідомлення або книга? Тільки дані, які повинні бути оброблені відповідно до правил (дата не може вказувати в майбутнє, e-mail повинен бути в певному форматі, ім'я не може бути довшим X символів, і так далі).
- Вид (View) - забезпечує різні способи представлення даних, які отримані з моделі. Він може бути шаблоном, який заповнюється даними. Може бути кілька різних видів, і контролер вибирає, який підходить якнайкраще для поточної ситуації.
- Контроллер (Controller) може бути влаштований як основний, який отримує всі запити і викликає інші контролери для виконання дій в залежності від ситуації.

Для моделі були обрані дані з БД (Речовини, Критичні органи, Фактор нахилу та інші). Які відображаються на веб сторінках програмного продукту. Зв'язок між серверною частиною та представлення зроблен через REST API.

REST (скор. англ. Representational State Transfer, «передача репрезентативного стану») — підхід до архітектури мережевих протоколів, які забезпечують доступ до інформаційних ресурсів. Дані повинні передаватися у вигляді невеликої кількості стандартних форматів (наприклад, HTML, XML, JSON). Будь-який REST протокол (HTTP в тому числі) повинен підтримувати кешування, не повинен залежати від мережевого прошарку, не повинен зберігати інформації про стан між парами «запит-відповідь». Стверджується, що такий підхід забезпечує масштабованість системи і дозволяє їй еволюціонувати з новими вимогами. [11]

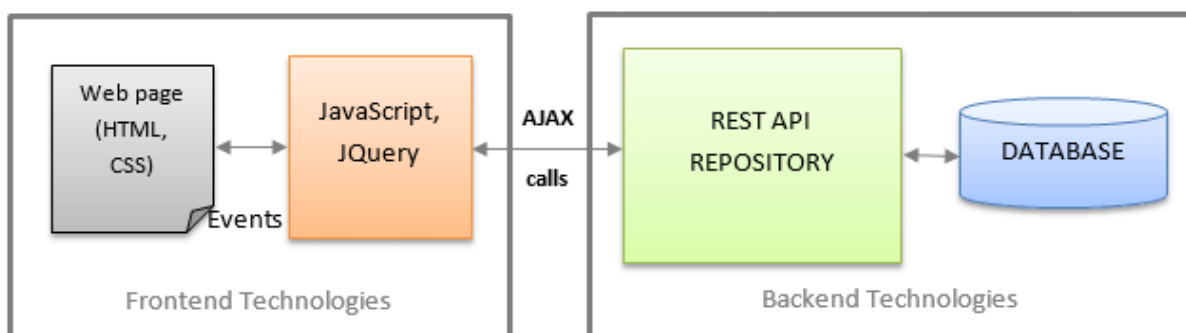


Рисунок 4.1 – Схема роботи REST.

Даний архітектурний стиль дозволяє:

- Мати клієнт-серверну архітектуру
- Не мати стану, тобто кожен запит містить всю необхідну інформацію для його обробки, і не покладається на те, що сервер знає щось з попереднього запиту.

Однорідний інтерфейс - Всі компоненти в архітектурі REST підтримують однорідний інтерфейс. Це зменшує зв'язність між компонентами і сервісами які вони надають і дозволяє нескладно змінювати компоненти при потребі.

БД розвертається у додатку Docker, який дозволяє інсталювати СУБД у власному додатку, та робити це незалежно від встановленої системи у користувача.

При першому запуску додатку виконується DML скрипт, який заповнює БД даними для подальшої маніпуляції з ними.

## 4.2 Use-Case діаграма

Розроблений веб додаток відповідає наступній use-case діаграмі, тобто опис поведінки додатку і як веб додаток відповідає на зовнішні запити (рисунок 4.2)

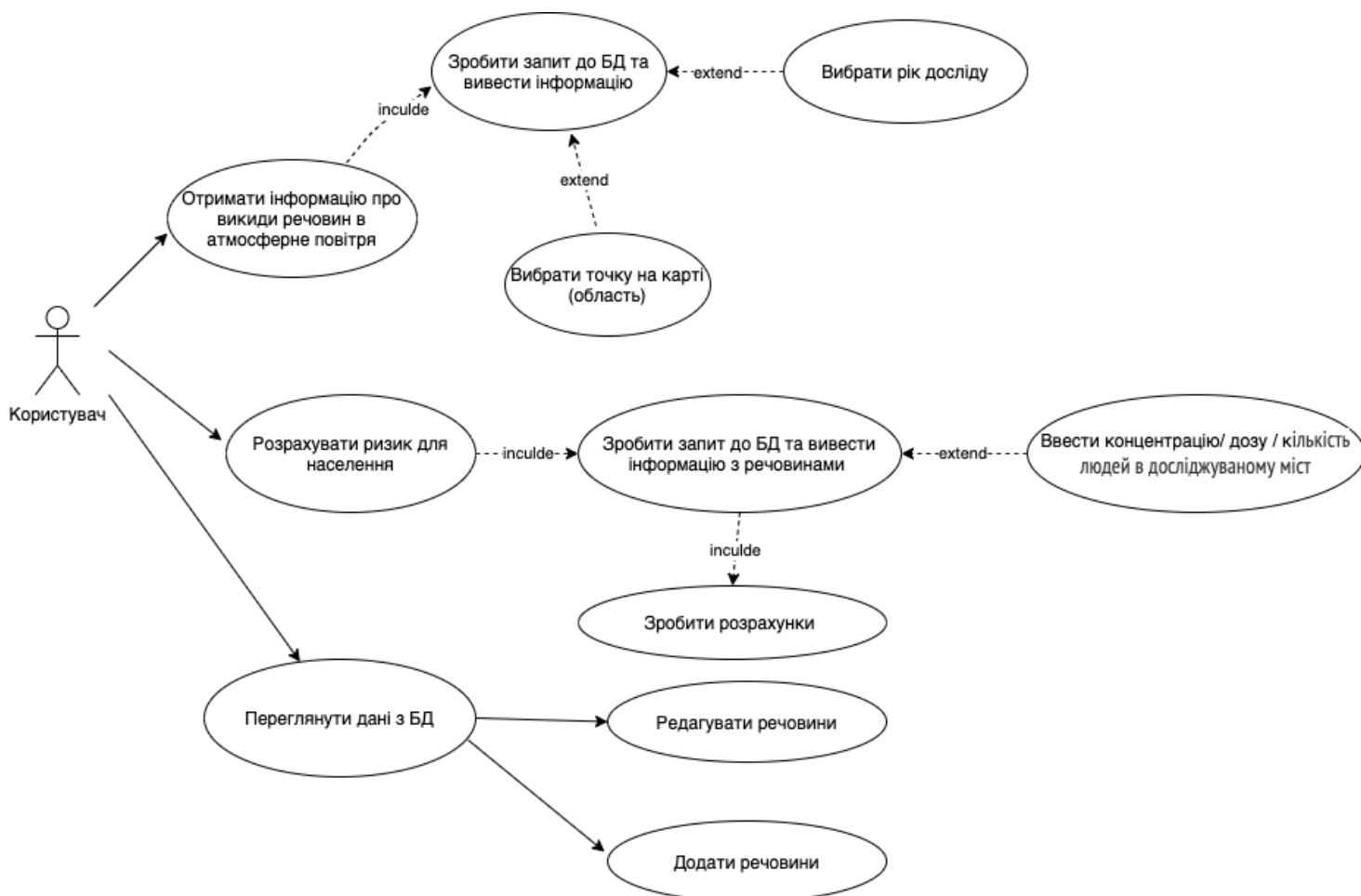


Рисунок 4.2 – Діаграма Use-case.

### 4.3 Концептуальна модель бази даних.

База даних складається з таблиць які відображують інформацію для розрахунку ризиків. Таблиці можуть різні зв'язки. Наприклад таблиця Substance має зв'язок Many – to – Many. Це тип, який відноситься до відносин між двома сутностями Substance і CriticalOrgan, в яких Substance може містити батьківський екземпляр, для якого в CriticalOrgan багато дітей, і навпаки маючи загальне представлення про сутності предметної області були сформовані окремі відповідні моделі (представлені у вигляді класів), що згодом за допомогою механізму міграції були трансформовані в таблиці БД (рис 4.3). Для представлення emission був використан зв'язок one-to-many з таблицею emission\_info для зберігання показників окремо по рокам. Зв'язок «один до багатьох» - це тип кардинальності, який відноситься до відносин між двома сутностями А і В, в яких елемент А може бути пов'язаний з багатьма елементами В, але членом В пов'язано лише з одним елементом А.

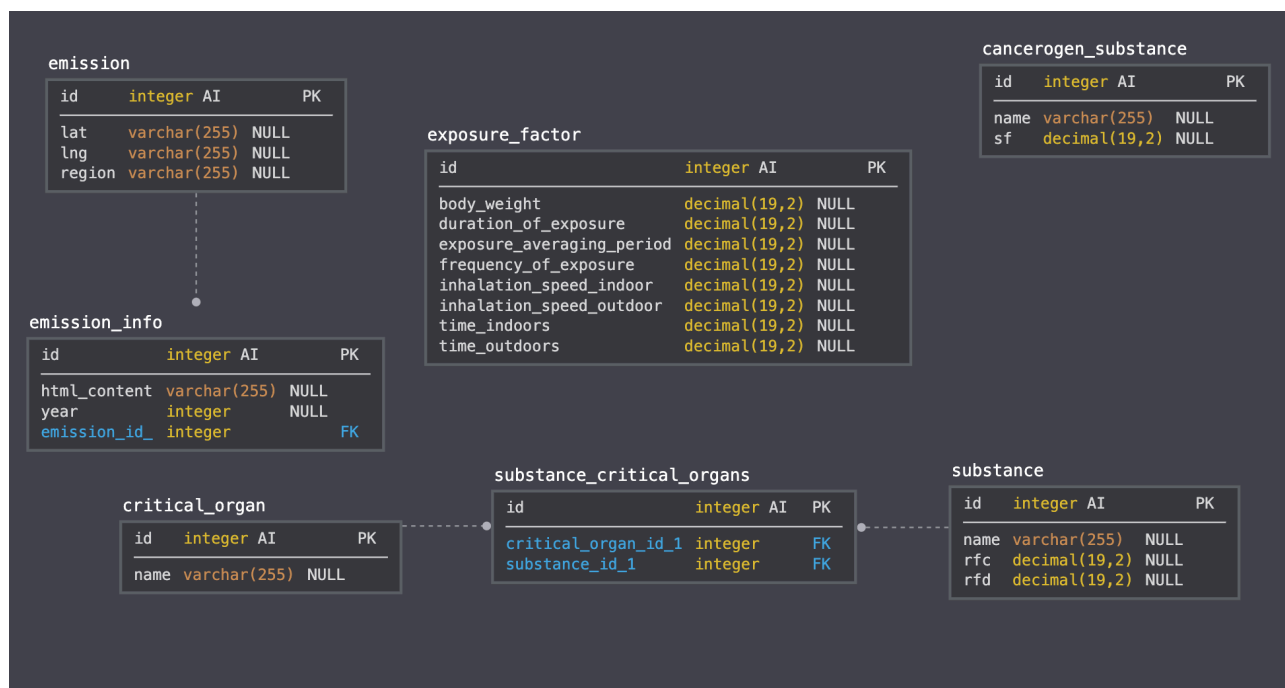


Рисунок 4.3 – Схема бази даних.

База даних розміщується в Docker. Docker - програмне забезпечення для автоматизації розгортання і управління додатками в середовищах з підтримкою контейнеризації. Дозволяє «упакувати» додаток з усім його оточенням і залежностями в контейнер, який може бути перенесений на будь-яку Linux-систему з підтримкою cgroups в ядрі, а також надає середовище з управління контейнерами. При використанні Docker розгортання БД стає простою задачею. На рисунку 4.4 команди для перегляду роботи контейнеру MySQL.

```
OlehS-MacBook-Air:~ olehdibrov$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
561e2207a877   mysql    "docker-entrypoint.s..."  8 days ago    Up 8 days    0.0.0.0:3306->3306/tcp, 33060/tcp   hb-mysql-example
OlehS-MacBook-Air:~ olehdibrov$ docker mysql logs
docker: 'mysql' is not a docker command.
See 'docker --help'
OlehS-MacBook-Air:~ olehdibrov$ docker logs mysql
Error: No such container: mysql
OlehS-MacBook-Air:~ olehdibrov$ docker logs hq-mysql-example
Error: No such container: hq-mysql-example
OlehS-MacBook-Air:~ olehdibrov$ docker logs 561e2207a877
2020-05-31 14:59:56+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.20-1debian10 started.
2020-05-31 14:59:56+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2020-05-31 14:59:56+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.20-1debian10 started.
2020-05-31 14:59:56+00:00 [Entrypoint]: Initializing database files
2020-05-31T14:59:56.352455Z 0 [Warning] [MY-011070] [Server] 'Disabling symbolic links using --skip-symbolic-links (or equivalent) is the default. Consider not using this option as it' is deprecated and will be removed in a future release.
2020-05-31T14:59:56.352614Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.20) initializing of server in progress as process 43
2020-05-31T14:59:56.360071Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2020-05-31T14:59:57.037075Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2020-05-31T14:59:58.912528Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option
```

Рисунок 4.4 – перегляду роботи контейнеру MySQL у Docker

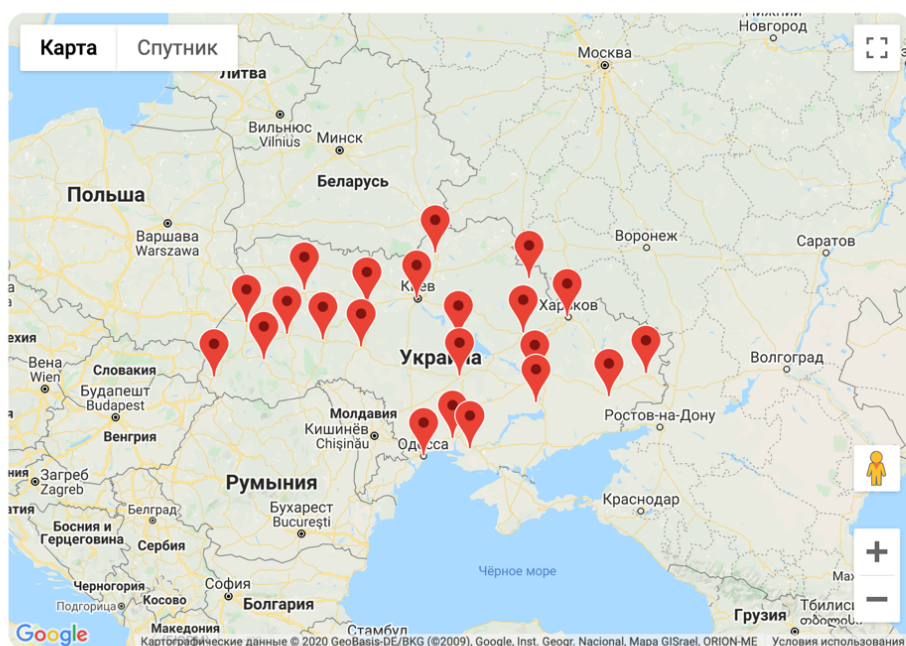
## 5. РОБОТА КОРИСТУВАЧА З СИСТЕМОЮ

Для запуску додатку потрібно веб сервер з java 8. Наступним кроком буде запуску jar файлу проекта. Або з IntelliJ IDEA запустити RiskAssesmentApplication.java (у IntelliJ IDEA є вбудований веб сервер Tomcat, який дозволяє запускати проекти). Наступним кроком є встановлення Docker та створення образу MySQL.

Після попадання на головну сторінку додатку (рис 5.1) користувач бачить інтерактивну карту України, маркери на карті та меню з можливостями додатку



Оцінка ризику здоров'я населення від забруднення атмосферного повітря



Оцінка ризику



Огляд речовин в БД

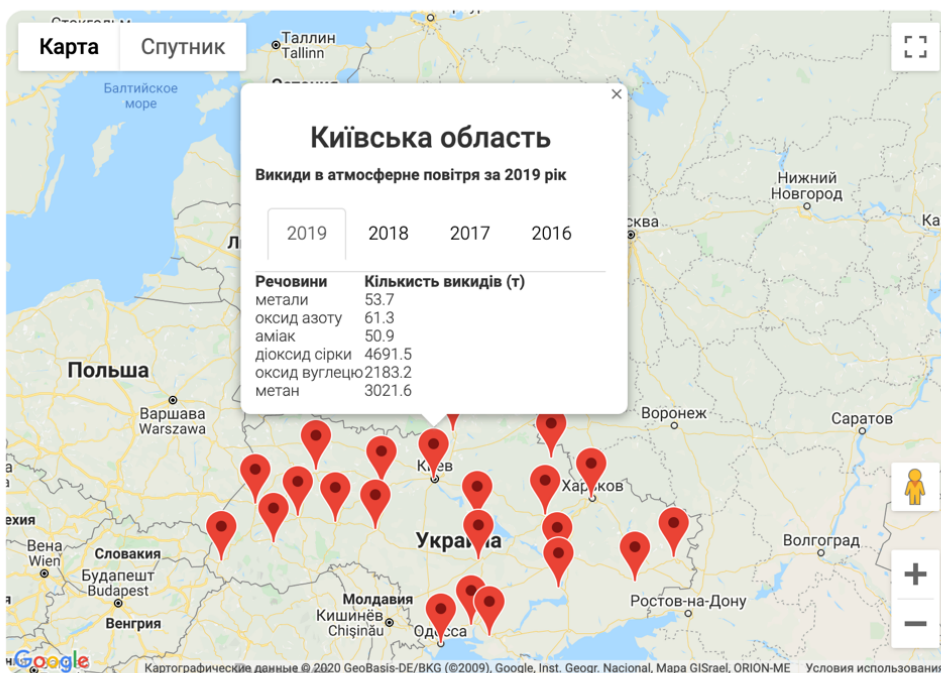
Copyright © 2020 | Powered by Oleg Dibrov

Рисунок 5.1 – Головна сторінка.

Карта містить маркери регіонів України. Користувач має можливість натиснути на один із них для отримання більш детальної інформації (рис 5.2).



## Оцінка ризику здоров'я населення від забруднення атмосферного повітря



Copyright © 2020 | Powered by Oleg Dibrov

Рисунок 5.2 – Інформація про викиди шкідливих речовин за 2019 рік.

Після вибору розділу «Оцінка ризику» користувач переходить на нову сторінку, де він може зробити аналіз оцінки ризику здоров'я населення від забруднення атмосферного повітря (рис 5.3)

Після відкриття головної сторінки розрахунку у користувача є 3 можливих розрахунків:

- Розрахунок неканцерогенного ризику
- Розрахунок сумарного неканцерогенного ризику (НІ)
- Розрахунок середню добової дози впливу речовини на населення міста)



## Оцінка ризику здоров'я населення від забруднення атмосферного повітря

Розрахунок неканцерогенного ризику

Розрахунок сумарного неканцерогенного ризику (HI)

Розрахунок середню добової дози впливу речовини на населення міста

Виберіть речовину

Азоту діоксид

Концентрація в атмосферному повітрі(мг/м3)

Розрахувати

Copyright © 2020 | Powered by Oleg Dibrov

?

Рисунок 5.3 – Головна сторінка розрахунку.



## Оцінка ризику здоров'я населення від забруднення атмосферного повітря

Розрахунок неканцерогенного ризику

Розрахунок сумарного неканцерогенного ризику (HI)

Розрахунок середню добової дози впливу речовини на населення міста

Виберіть речовину

Азоту діоксид

Концентрація в атмосферному повітрі(мг/м3)

0.0003


Розрахувати

Коефіцієнт небезпеки (HQ) = 0.0060 Ризик виникнення шкідливих ефектів розглядають як зневажаєливо малий

Copyright © 2020 | Powered by Oleg Dibrov

Рисунок 5.4 – Приклад розрахунку неканцерогенного ризику

У додатка є можливість розрахування сумарного неканцерогенного ризику. Для цього користувач спочатку вибирає досліджувані речовини. На основі цих речовин будується таблиця речовин. Наступним кроком є введення доз(мг/кг) для кожної речовини. Результат будується в сумарну таблицю (рис. 5.5)

 **Оцінка ризику здоров'я населення від забруднення атмосферного повітря**

Розрахунок неканцерогенного ризику   
 Розрахунок сумарного неканцерогенного ризику (НІ)   
 Розрахунок середню добової дози впливу речовини на населення міста

---

Виберіть речовину Речовина не вибрана Побудувати таблицю

Речовина	Доза, мг/кг	RfD, мг/кг	HQ	Критичні органи
Азоту діоксид	<input type="text" value="0.002"/>	0.088000	0.02273	Органи дихання, ЦНС
Азотна кислота	<input type="text" value="0.0003"/>	0.060000	0.00500	Органи дихання
Акрилова к-та	<input type="text" value="0.0001"/>	0.002000	0.05000	Органи дихання

Розрахувати

Сумарний ризик	НІ загальний	0.10046
	НІ Органи дихання	0.07773
	НІ ЦНС	0.02273

Copyright © 2020 | Powered by Open Eklon

Рисунок 5.4 – Приклад розрахунку сумарного неканцерогенного ризику

Для швидкого оброблення інформацію користувачем у нижньому правому кутку є кнопка допомоги для розшифрування ризику здоров'я. Така кнопка є на кожній з трьох задач оцінки ризику (рисунок 5.5)

Критерії неканцерогенного ризику

Характеристика ризику	Коефіцієнт небезпеки (HQ)
Ризик виникнення шкідливих ефектів розглядають як зневажливо малий	< 1
Гранична величина, що не потребує термінових заходів, однак не може розглядатися як досить прийнятна	1
Імовірність розвитку шкідливих ефектів зростає пропорційно збільшенню HQ	> 1

Закрити

Розрахунок неканцерогенного ризику

Виберіть речовину

Речовина

Азоту діоксид

Азотна кислота

Акрилова к-та

Розрахувати

Сумарний ризик	НІ загальний	0.10046
	НІ Органи дихання	0.07773
	НІ ЦНС	0.02273

Рисунок 5.5 – вікно допомоги при розрахунку ризику

Оцінка ризику здоров'я населення від забруднення атмосферного повітря

Розрахунок неканцерогенного ризику    Розрахунок сумарного неканцерогенного ризику (НІ)    Розрахунок середню добової дози впливу речовини на населення міста

Виберіть канцерогенову речовину

Азобензол

Концентрація в атмосферному повітрі(мг/м3)

0.01

Кількість людей в досліджуваному місті

300000

Розрахувати

Величина індивідуального ризику: 0.00046224657534241 . Величина популяційного ризику: 138.67397260272300000 . Класифікація рівня ризику - Середній – припустимий для виробничих умов; за впливу на все населення необхідний динамічний контроль і поглиблене вивчення джерел і можливих наслідків шкідливих впливів для вирішення питання про заходи з управління ризиком

Рисунок 5.5 – Вкладка розрахуноку середню добової дози впливу речовини на населення міста (канцерогенних речовин)

У користувача ще є можливість огляду БД. А саме додавати та робити зміни в речовинах (Канцерогенів та неканцерогенів)



## Оцінка ризику здоров'я населення від забруднення атмосферного повітря

Неканцерогенні речовини

Концерогенні речовини

id	Речовина	RfC, мг/м3	RfD, мг/кг	Критичні органи	Дія
1	Азоту діоксид	<input type="text" value="0,050000"/>	<input type="text" value="0,088000"/>	Органи дихання, ЦНС	<input type="button" value="Зберегти"/>
2	Азотна кислота	<input type="text" value="0,040000"/>	<input type="text" value="0,060000"/>	Органи дихання	<input type="button" value="Зберегти"/>
3	Акрилова к-та	<input type="text" value="0,002000"/>	<input type="text" value="0,002000"/>	Органи дихання	<input type="button" value="Зберегти"/>
4	Акрилонітрил	<input type="text" value="0,004500"/>	<input type="text" value="0,000110"/>	Органи дихання	<input type="button" value="Зберегти"/>
5	Акролеїн	<input type="text" value="0,006000"/>	<input type="text" value="0,000400"/>	Органи дихання	<input type="button" value="Зберегти"/>
6	Речовина	<input type="text" value="0,001000"/>	<input type="text" value="0,000200"/>	Органи дихання, ЦНС, Селезінка	<input type="button" value="Зберегти"/>
	<input type="text" value="Речовина"/>	<input type="text" value="0.01"/>	<input type="text" value="0.01"/>	<input type="text" value="Критичні органи не вибрані"/> <input type="button" value="+"/>	<input type="button" value="Додати"/>

Рисунок 5.6 – Зміни та додавання неканцерогенних речовин



## Оцінка ризику здоров'я населення від забруднення атмосферного повітря

Неканцерогенні речовини

Концерогенні речовини

id	Речовина	SF, мг/(кг × доба) <sup>-1</sup>	Дія
1	Азасерин	<input type="text" value="11,00"/>	<input type="button" value="Зберегти"/>
2	Азатіоприн	<input type="text" value="1,80"/>	<input type="button" value="Зберегти"/>
3	Азобензол	<input type="text" value="0,37"/>	<input type="button" value="Зберегти"/>
4	Речовина	<input type="text" value="0,00"/>	<input type="button" value="Зберегти"/>
	<input type="text" value="Речовина"/>	<input type="text" value="0.01"/>	<input type="button" value="Додати"/>

Рисунок 5.7 – Зміни та додавання неканцерогенних речовин

# ВИСНОВКИ

При виконанні дипломної роботи була досліджена тема оцінка ризику здоров'я населення від забруднення атмосферного повітря. Був здобут досвід у розробці веб додатків з інтерактивною картою, а також досвід у проектуванні та розгортанні баз даних на сервер. Робота з технологіями Spring, Spring Boot, Web та Docker.

Застосування системи оцінки ризику дозволить забезпечити: безперервний збір даних концентрацій забруднюючих речовин та викидів в атмосферу, обробку отриманих даних за допомогою програмного забезпечення, зберігання та аналіз інформації про стан навколишнього природного середовища, прогнозування його змін та розробку рекомендацій для прийняття управлінських рішень.

Система може працювати не тільки на певних територіях країни, а й по всій Україні, що допоможе значно скоротити час на проведення лабораторних досліджень.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Н. В. Караєва, І. А. Варава. Методи і засоби оцінки ризику здоров'ю населення від забруднення атмосферного повітря – Київ, 2018 – 25 с.
2. Методичні рекомендації МР 2.2.12-142-2007. Оцінка ризику для здоров'я населення від забруднення атмосферного повітря. Затв. Наказом МОЗ України від 13.04.07 № 184. – Київ, 2007. – 40 с.
3. Karaieva N. V. Methodology of social and environmental external costs estimation in the Ukraine's energy sector [Online] / N. V. Karaieva, G. V. Kramarev // Economic Processes Management: International Scientific E-Journal. №3 – 2016. – Available: [http://epm.fem.sumdu.edu.ua/download/2016\\_3/epm\\_2016\\_3.pdf](http://epm.fem.sumdu.edu.ua/download/2016_3/epm_2016_3.pdf).
4. Порядок декларування безпеки об'єктів підвищеної небезпеки. Затверджено постановою Кабінету Міністрів України від 11.07.2002р. №956. ДНАОП 0.00-8.22-02
6. Джошуа Блох Java: эффективное программирование Третье издание – 43 с.
7. Документація по Spring [Електронний ресурс] – Режим доступу до ресурсу: <https://spring.io/projects/spring-framework>
8. Документація по Spring Boot [Електронний ресурс] – Режим доступу до ресурсу: <https://spring.io/projects/spring-boot>
9. Документація по MySQL [Електронний ресурс] – Режим доступу до ресурсу: <http://www.mysql.ru/docs/mysql-man-4.0-ru/>
10. Документація по Docker [Електронний ресурс] – Режим доступу до ресурсу: <https://dker.ru/docs/docker-engine/get-started-with-docker/get-started-with-docker/>
11. Документація по Spring REST [Електронний ресурс] – Режим доступу до ресурсу: <https://spring.io/projects/spring-restdocs>

# ДОДАТОК 1

Веб додаток для оцінки ризику здоров'я населення від забруднення  
атмосферного повітря

Специфікація

УКР.НТУУ"КПІ ім. Ігоря Сікорського" \_ТЕФ\_АПЕПС\_ТМ62196\_20Б

Аркушів 2

Київ 2020

Позначення	Найменування	Примітки
Документація		
УКР.НТУУ«КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62196_20Б	Записка.docx	Пояснювальна записка
Компоненти		
УКР.НТУУ«КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62196_20Б 12-1	src/main/resources/application.properties	Модуль конфігурації
УКР.НТУУ«КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62196_20Б 12-2	src/main/java/service	Модулі компонентів
УКР.НТУУ«КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62196_20Б 12-3	src/main/resources/templates/calculator.html	Модуль графічного інтерфейсу
УКР.НТУУ«КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62196_20Б 13-1	Опис.docx	Опис програми

# ДОДАТОК 2

Веб додаток для оцінки ризику здоров'я населення від забруднення  
атмосферного повітря

Текст програми

УКР.НТУУ” КПІ ім. Ігоря Сікорського ”\_ТЕФ\_АПЕПС\_ТМ62196\_20Б 12

Аркушів 14

Київ 2020

## УКР.НТУУ” КПІ ім. Ігоря Сікорського ”\_ТЕФ\_АПЕПС\_ТМ62196\_20Б 12-1

```
spring.jpa.hibernate.ddl-auto=update
spring.datasource.url=jdbc:mysql://localhost:3306/mysql
spring.datasource.username=root
spring.datasource.password=Buster

spring.jpa.database-platform=org.hibernate.dialect.MySQL5InnoDBDialect
spring.jpa.properties.hibernate.id.new_generator_mappings=false
spring.jpa.generate-ddl=true
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.format_sql=true

spring.thymeleaf.cache=false
```

## УКР.НТУУ” КПІ ім. Ігоря Сікорського ”\_ТЕФ\_АПЕПС\_ТМ62196\_20Б 12-2

```
package com.kpi.diplom.riskassessment.controller;

import com.kpi.diplom.riskassessment.domain.Substance;

import com.kpi.diplom.riskassessment.repository.CancerogenSubstanceRepository;

import com.kpi.diplom.riskassessment.repository.SubstanceRepository;

import com.kpi.diplom.riskassessment.service.SubstanceService;

import com.kpi.diplom.riskassessment.service.calculator.risk.AverageDailyDoseCalculator;

import com.kpi.diplom.riskassessment.service.calculator.risk.NonCarcinogenicCalculator;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.bind.annotation.ResponseBody;

import java.util.ArrayList;

import java.util.List;
```

@Service

```

public class AverageDailyDoseCalculator {

    Range<BigDecimal> high = Range.between(new BigDecimal("0.001"), new BigDecimal("10000000"));

    Range<BigDecimal> medium = Range.between(new BigDecimal("0.0001"), new BigDecimal("0.001"));

    Range<BigDecimal> low = Range.between(new BigDecimal("0.000001"), new BigDecimal("0.0001"));

    Range<BigDecimal> min = Range.between(new BigDecimal("0.000000001"), new BigDecimal("0.000001"));

    @Autowired

    private ExposureFactorRepository exposureFactorRepository;

    @Autowired

    private CancerogenSubstanceRepository cancerogenSubstanceRepository;

    public String calcAverageDailyDose(String substance, String concentration, String numOfPeoples) {

        ExposureFactor exposureFactor = exposureFactorRepository.findById(1).get();

        BigDecimal conc = new BigDecimal(concentration);

        BigDecimal left =
conc.multiply(exposureFactor.getTimeOutdoors()).multiply(exposureFactor.getInhalationSpeedOutdoor());

        BigDecimal right =
conc.multiply(exposureFactor.getTimeIndoors()).multiply(exposureFactor.getInhalationSpeedIndoor());

        BigDecimal adding = left.add(right);

        BigDecimal multiply =
adding.multiply(exposureFactor.getFrequencyOfExposure()).multiply(exposureFactor.getDurationOfExposure());

        BigDecimal bottom =
exposureFactor.getBodyWeight().multiply(exposureFactor.getExposureAveragingPeriod()).multiply(BigDecimal.valueOf(365));

        BigDecimal result = multiply.divide(bottom, 15, RoundingMode.HALF_UP);

        CancerogenSubstance cancerogenSubstance = cancerogenSubstanceRepository.findByName(substance);

        BigDecimal cr = result.multiply(cancerogenSubstance.getSf());

        BigDecimal pcr = cr.multiply(new BigDecimal(numOfPeoples));

        return formMessage(cr, pcr);

    }

    private String formMessage(BigDecimal result, BigDecimal pcr) {

        if (high.contains(result)) {

```

```

        return "Величина індивідуального ризику: " + result.toString() + "\n. Величина популяційного
ризику: " + pcr.toString() + "\n. Класифікація рівня ризику - Високий (De Manifestis) – не прийнятний для
виробничих умов і населення. Необхідне здійснення заходів з усунення або зниження ризику";

    } else {

        if (medium.contains(result)) {

            return "Величина індивідуального ризику: " + result.toString() + "\n. Величина популяційного
ризику: " + pcr.toString() + "\n. Класифікація рівня ризику - Середній – припустимий для виробничих умов; за
впливу на все населення необхідний динамічний контроль і поглиблене вивчення джерел і можливих наслідків
шкідливих впливів для вирішення питання про заходи з управління ризиком";

        } else {

            if (low.contains(result)) {

                return "Величина індивідуального ризику: " + result.toString() + "\n. Величина
популяційного ризику: " + pcr.toString() + "\n. Класифікація рівня ризику - Низький – припустимий ризик
(рівень, на якому, як правило, встановлюються гігієнічні нормативи для населення)";

            } else {

                return "Величина індивідуального ризику: " + result.toString() + "\n. Величина
популяційного ризику: " + pcr.toString() + "\n. Класифікація рівня ризику - Мінімальний (De Minimis) –
бажана (цільова) величина ризику при проведенні оздоровчих і природоохоронних заходів";

            }

        }

    }

}

```

```
package com.kpi.diplom.riskassessment.service.calculator.risk;
```

```
import com.kpi.diplom.riskassessment.domain.CriticalOrgan;
```

```
import com.kpi.diplom.riskassessment.domain.Substance;
```

```
import com.kpi.diplom.riskassessment.domain.SubstanceCriticalOrgans;
```

```
import com.kpi.diplom.riskassessment.repository.SubstanceRepository;
```

```
import com.kpi.diplom.riskassessment.service.SubstanceService;
```

```
import org.json.JSONObject;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Service;
```

```
import java.math.BigDecimal;
```

```
import java.math.RoundingMode;
```

```
import java.util.Collection;
```

```
import java.util.List;
```

```

import java.util.Map;

import java.util.stream.Collectors;

@Service

public class NonCarcinogenicCalculator {

    @Autowired

    private SubstanceRepository substanceRepository;

    @Autowired

    private SubstanceService substanceService;

    public String calculateRisk(String substanceName, String concentration) {

        Substance substance = substanceRepository.findByName(substanceName);

        return substance != null ? calc(substance, concentration) : "Not valid data received. Try again!";

    }

    private String calc(Substance substance, String concentration) {

        BigDecimal c = new BigDecimal(concentration);

        BigDecimal result = c.divide(substance.getRfc(), 5, RoundingMode.HALF_UP);

        return formMessage(result);

    }

    private String formMessage(BigDecimal result) {

        int resultOfComparing = result.compareTo(BigDecimal.ONE);

        switch (resultOfComparing) {

            case -1:

                return "Коефіцієнт небезпеки (HQ) = " + result.setScale(4, RoundingMode.HALF_UP).toString() +

                    "\nРизик виникнення шкідливих ефектів розглядають як зневажливо малий";

            case 0:

                return "Коефіцієнт небезпеки (HQ) = " + result.setScale(4, RoundingMode.HALF_UP).toString() +

                    "\nГранична величина, що не потребує термінових заходів, однак не може розглядатися як досить прийнятна";

            case 1:

                return "Коефіцієнт небезпеки (HQ) = " + result.setScale(4, RoundingMode.HALF_UP).toString() +

                    "\nІмовірність розвитку шкідливих ефектів зростає пропорційно збільшенню HQ";

        }

    }

}

```

```

        default:

            return "Not valid data received. Try again!";

    }
}

public String calculateSumRisk(List<String> substances, List<String> doses) {

    JSONObject jsonString = new JSONObject();

    List<Substance> chosenSubstances = substanceService.getChosenSubstancesById(substances);

    List<CriticalOrgan> criticalOrgans = chosenSubstances.stream()

        .map(Substance::getCriticalOrgans)

        .flatMap(Collection::stream)

        .map(SubstanceCriticalOrgans::getCriticalOrgan)

        .distinct()

        .collect(Collectors.toList());

    Map<String, BigDecimal> organsMap = criticalOrgans.stream()

        .collect(Collectors.toMap(CriticalOrgan::getName, x -> BigDecimal.ZERO));

    JSONObject hqObject = new JSONObject();

    jsonString.put("HQ", hqObject);

    BigDecimal hqGeneral = BigDecimal.ZERO;

    for (int i = 0; i < chosenSubstances.size(); i++) {

        Substance substance = chosenSubstances.get(i);

        BigDecimal dose = new BigDecimal(doses.get(i));

        BigDecimal result = dose.divide(substance.getRfd(), 5, RoundingMode.HALF_UP);

        hqObject.put("hq" + substance.getId(), result.toString());

        for (SubstanceCriticalOrgans x : substance.getCriticalOrgans()) {

            BigDecimal result1 = organsMap.get(x.getCriticalOrgan().getName()).add(result);

            organsMap.put(x.getCriticalOrgan().getName(), result1);

            hqGeneral = hqGeneral.add(result);

        }

    }

    jsonString.put("hqGeneral", hqGeneral.toString());
}

```

```

        JSONObject hqObjectOrgans = new JSONObject();

        jsonString.put("HQorgans", hqObjectOrgans);

        organsMap.forEach((key, value) -> hqObjectOrgans.put(key, value.toString()));

        return jsonString.toString();
    }
}

```

## УКР.НТУУ” КПІ ім. Ігоря Сікорського ”\_ТЕФ\_АПЕПС\_ТМ62196\_20Б 12-3

```

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

<head>

    <title>Home</title>

    <meta name="description" content="spring-boot thymeleaf | home ">

    <div th:replace="fragments/header :: header-css"></div>

    <script src="/js/jquery-3.5.1.min.js"></script>

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

    <link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" rel="stylesheet"/>

    <link rel="stylesheet"

        href="https://cdn.jsdelivr.net/npm/bootstrap-select@1.13.14/dist/css/bootstrap-select.min.css">

    <script src="https://cdn.jsdelivr.net/npm/bootstrap-select@1.13.14/dist/js/bootstrap-
select.min.js"></script>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap-select@1.13.14/dist/js/i18n/defaults-
*.min.js"></script>

    <link rel="stylesheet" type="text/css" th:href="@{/css/site.css}"/>

</head>

<body>

<div class="container">

    <div class="row header">

        <div class="row header-row">

            <a href="/">

```

```

        <h2>Оцінка ризику здоров'я населення від забруднення атмосферного повітря</h2>

    </a>

</div>

</div>

<!-- Nav tabs -->

<ul class="nav nav-tabs" id="myTab" role="tablist">

    <li class="nav-item">

        <a class="nav-link active" id="home-tab" data-toggle="tab" href="#home" role="tab" aria-
controls="home"

            aria-selected="true">Розрахунок неканцерогенного ризику</a>

    </li>

    <li class="nav-item">

        <a class="nav-link" id="profile-tab" data-toggle="tab" href="#profile" role="tab" aria-
controls="profile"

            aria-selected="false">Розрахунок сумарного неканцерогенного ризику (HI)</a>

    </li>

    <li class="nav-item">

        <a class="nav-link" id="messages-tab" data-toggle="tab" href="#messages" role="tab" aria-
controls="messages"

            aria-selected="false">Розрахунок середню добової дози впливу речовини на населення міста</a>

    </li>

</ul>

<!-- Tab panes -->

<div class="tab-content container-of-forms">

    <div class="tab-pane active" id="home" role="tabpanel" aria-labelledby="home-tab">

        <form id="non-conc-form" method="post" th:action="@{/calculate-non-conc-assessment}">

            <div class="form-group">

                <label for="substance">Виберіть речовину</label>

                <select class="form-control" id="substance" th:name="substance">

                    <th:block th:each="substance : ${substances}">

                        <option th:text="${substance.name}"></option>

                    </th:block>

                </select>

            </div>

            <div class="form-group">

                <label for="concentration">Концентрація в атмосферному повітрі(мг/м3)</label>

```

```

        <textarea class="form-control text-area" id="concentration" rows="3"
th:name="concentration"></textarea>

    </div>

    <div class="form-group row">

        <div class="col-sm-10">

            <button type="submit" class="btn btn-primary">Разрахувати</button>

        </div>

    </div>

</form>

<div id="result-of-assessment1" class="row"></div>

<!-- info button-->

<div class="box">

    <button type="button" class="box btn btn-primary" data-toggle="modal" data-
target="#exampleModal" ?

    </button>

</div>

<!-- Modal -->

<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-
labelledby="exampleModallabel"

    aria-hidden="true">

    <div class="modal-dialog" role="document">

        <div class="modal-content">

            <div class="modal-header">

                <h5 class="modal-title" id="exampleModallabel">Критерії неканцерогенного
ризик</h5>

                <button type="button" class="close" data-dismiss="modal" aria-label="Close">

                    <span aria-hidden="true">&times;</span>

                </button>

            </div>

            <div class="modal-footer">

                <button type="button" class="btn btn-secondary" data-
dismiss="modal">Закрити</button>

            </div>

        </div>

    </div>

</div>

```

```

</div>

<div class="tab-pane" id="profile" role="tabpanel" aria-labelledby="profile-tab">
  <form method="post" id="fill-table" th:action="@{/getChosenSubstancesInfo}">
    <label for="chosen">Виберіть речовину</label>
    <select id="chosen" class="selectpicker" title="Речовина не вибрана" multiple data-live-search="true" th:name="chosen">
      <th:block th:each="substance : ${substances}">
        <option th:text="${substance.name}"></option>
      </th:block>
    </select>
    <button type="submit" class="btn btn-primary">Побудувати таблицю</button>
  </form>

  <form method="post" id="non-conc-form-amount" th:action="@{/calculate-amount-of-non-conc-assessment}">
    <table class="table">
      <thead>
        <tr class="tableizer-firstrow">
          <th>Речовина</th>
          <th>Доза, мг/кг</th>
          <th>RfD, мг/кг</th>
          <th>HQ</th>
          <th>Критичні органи</th>
        </tr>
      </thead>
      <tbody>
        <th:block th:each="substance2 : ${chosenSubstances}">
          <tr>
            <td th:text="${substance2.name}"></td>
            <td>
              <div class="input-group mb-3">
                <input type="number" class="form-control" name="dose" placeholder="0.01"
                  step="0.00000001" aria-label="dose"
                  aria-describedby="basic-addon1">
                <input type="hidden" class="form-control" name="id"
                  th:value="${substance2.id}">

```

```

        </div>

    </td>

    <td th:text="{substance2.rfd}"></td>

    <td th:id="'hq' + {substance2.id}"></td>

    <td th:text="{substance2.toStringCriticalOrgans()}"></td>

</tr>

</th:block>

</tbody>

</table>

<div th:if="{btnCalculate}">

    <button type="submit" id="btn-calculate-true" class="btn btn-
primary">Розрахувати</button>

</div>

<!-- ELSE -->

<div th:unless="{btnCalculate}">

    <button type="submit" id="btn-calculate-false" disabled class="btn btn-
primary">Розрахувати</button>

</div>

</form>

<div id="result-of-assessment2" class="row"></div>

<!-- info button-->

<div class="box">

    <button type="button" class="box btn btn-primary" data-toggle="modal" data-
target="#exampleModal2">?

    </button>

</div>

</div>

<div class="tab-pane" id="messages" role="tabpanel" aria-labelledby="messages-tab">

<form id="average-daily-dose-form" method="post" th:action="@{/average-daily-dose}">

    <div class="form-group">

        <label for="substance">Виберіть канцерогенову речовину</label>

        <select class="form-control" th:name="substance">

            <th:block th:each="substance : {cancerogenSubstances}">

                <option th:text="{substance.name}"></option>

```

```
        </th:block>

    </select>

</div>

<div class="form-group">

    <label for="concentration1">Концентрація в атмосферному повітрі(мг/м3)</label>

    <textarea class="form-control text-area" id="concentration1" rows="3"
th:name="concentration1"></textarea>

</div>

<div class="form-group">

    <label for="numOfPeoples">Кількість людей в досліджуваному місті</label>

    <textarea class="form-control text-area" id="numOfPeoples" rows="3"
th:name="numOfPeoples"></textarea>

</div>

<div class="form-group row">

    <div class="col-sm-10">

        <button type="submit" class="btn btn-primary">Разрахувати</button>

    </div>

</div>

</form>

<div id="result-of-assessment3" class="row"></div>

<!-- Modal -->

<div class="modal fade" id="exampleModal1" tabindex="-1" role="dialog" aria-
labelledby="exampleModallabel1"

    aria-hidden="true">

    <div class="modal-dialog" role="document">

</div>

</div>

</div>

<footer>

    <div class="separator" th:replace="fragments/footer :: footer"></div>

</footer>

</body>

<script>
```

```

(function ($) {
    if (document.location.href.indexOf('getChosenSubstancesInfo') > -1) {

        $("a[href='" + "#profile" + "']").tab("show");

    }
})(jQuery);

(function ($) {
    $(".text-area").keydown(function (event) {

        if (event.shiftKey == true) {

            event.preventDefault();

        }

        if ((event.keyCode >= 48 && event.keyCode <= 57) ||

            (event.keyCode >= 96 && event.keyCode <= 105) ||

            event.keyCode == 8 || event.keyCode == 9 || event.keyCode == 37 ||

            event.keyCode == 39 || event.keyCode == 46 || event.keyCode == 190) {

        } else {

            event.preventDefault();

        }

        if($(this).val().indexOf('.') !== -1 && event.keyCode == 190)

            event.preventDefault();

    });
})(jQuery);

(function ($) {

    var $form = $('#non-conc-form');

    $form.on('submit', function (e) {

        e.preventDefault();

        $.ajax({

            url: $form.attr('action'),

            type: 'post',

```

```
        data: $form.serialize(),
        success: function (response) {
            console.log(response);
            $('#result-of-assessment1').text(response);
        }
    });
})
}(jQuery));
```

```
(function ($) {
    var $form = $('#average-daily-dose-form');
    $form.on('submit', function (e) {
        e.preventDefault();
        $.ajax({
            url: $form.attr('action'),
            type: 'post',
            data: $form.serialize(),
            success: function (response) {
                console.log(response);
                $('#result-of-assessment3').text(response);
            }
        });
    })
}(jQuery));
```

```
(function ($) {
    var $form = $('#non-conc-form-amount');
    $form.on('submit', function (e) {
        e.preventDefault();
        $.ajax({
            url: $form.attr('action'),
            type: 'post',
            data: $form.serialize(),
            success: function (response) {
                console.log(response);
            }
        });
    })
}(jQuery));
```

```

        var json = JSON.parse(response);

        var result = parseJsonResultAndCreateHtml(json);

        console.log(result);

        $('#result-of-assessment2').html(result);

        for (var key in json.HQ) {

            $('##' + key).text(json.HQ[key]);

        }

    }

    });

    })

})(jQuery));

function parseJsonResultAndCreateHtml(jsonResult) {

    return "<table class=\"table\"> " +

        "<thead><tr><th>Сумарний ризик</th><th>НІ загальний</th><th>" + jsonResult["hqGeneral"] +

"</th></tr></thead><tbody>" + getHtmlByOrgans(jsonResult) + "</tbody></table>";

    }

function getHtmlByOrgans(jsonResult) {

    var string = "";

    for (var key in jsonResult.HQorgans) {

        string += "<tr><td></td><td>" + "НІ " + key + "</td><td>" + jsonResult.HQorgans[key] +

"</td></tr>";

    }

    return string;

    }

</script>

</html>

```

# ДОДАТОК 3

Веб додаток для оцінки ризику здоров'я населення від забруднення  
атмосферного повітря

Опис програми

УКР.НТУУ"КПІ ім. Ігоря Сікорського" \_ТЕФ\_АПЕПС\_ТМ62196\_20Б 13-1

Аркушів 8

Київ – 2020

## АНОТАЦІЯ

Веб додаток дає можливість оцінювати ризик здоров'я населення від забруднення атмосферного повітря. В цьому програмному продукті реалізовано графічний користувацький інтерфейс для легкої взаємодії з нею на будь-якому комп'ютері чи смартфоні. Додаток обробляє запити користувача, та виводить на екран результати оцінки ризику здоров'я. Також користувач може створювати та редагувати об'єкти в БД.

Застосунок розроблено в середовищі розробки JetBrains IntelliJ IDEA 2019 з використанням мови програмування Java, та з базою даних MySQL в контейнері Docker для легкого розгортання.

## ЗМІСТ

1. Загальні відомості.....	4
2. Функціональне призначення .....	5
3. Опис логічної структури.....	6
4. Використовувані технічні засоби.....	7
5. Вхідні і вихідні дані.....	8

## ЗАГАЛЬНІ ВІДОМОСТІ

Розроблений програмний веб додаток містить серверну частину написану на мові Java у зв'язці з базою даних MySQL. Керування базою даних знаходиться в контейнері Docker. Сторінки веб інтерфейсу тісно зв'язані з серверною частиною, бо йде запит до Java серверу, який дістає та обробляє дані і передає їх клієнту на HTML сторінку.

## **ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ**

Розроблена програмна система створена для оцінки ризику здоров'я населення від забруднення атмосферного повітря. Додаток може використовуватися виконання дослідження розвитку негативних ефектів на здоров'ї людини.

Мета модуля для роботи з базою даних - збереження інформації від користувача веб додатка, а також обмін інформацією з клієнтським інтерфейсом. Використання такого шаблону дозволяє створити програмне забезпечення, в якому інтерфейс і логіка модуля для бази даних є незалежними компонентами, що дозволяє використовувати його для зменшення навантаження на стороні клієнта системи.

# ОПИС ЛОГІЧНОЇ СТРУКТУРИ

Розроблена програмна система складається з декількох компонентів:

- Серверна частина розроблена на Java ;
- Клієнтська частина HTML, CSS та JS
- Контейнер з базою даних MySQL всередині, для розгортання БД

Взаємодія всіх вказаних вище компонентів між собою надає можливість коректного виконання програми.

## **ВИКОРИСТОВУВАНІ ТЕХНІЧНІ ЗАСОБИ**

Програмний продукт реалізований як окремий Java клас, який забезпечує існування клієнтської частини та серверної окремо від одного, але запуск обох компонентів відбувається одночасно. Для використання програмного продукту необхідний пристрій з встановленою Java 8.

## **ВХІДНІ І ВИХІДНІ ДАНІ**

Вхідними даними для програмного продукту є інформація, яку користувач вводить в веб додатку.

Вихідними даними програмного продукту є необхідні результати, які потрібні для користувача.