

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

« ____ » _____ 2021 р.

Дипломна робота

на здобуття ступеня бакалавра

спеціальності 122 «Комп'ютерні науки»

освітня програма «Комп'ютерний моніторинг та геометричне моделювання процесів і систем»

на тему: «Система моніторингу рівня глобальних ризиків»

Виконав:

студент IV курсу, групи ТМ-72

Лесик Максим Олександрович _____

Керівник:

Доцент, к.е.н.,

Караєва Наталія Веніамінівна _____

Рецензент:

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

Студент _____

Київ — 2021 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

Спеціалізація: Комп'ютерний моніторинг та геометричне моделювання процесів і систем

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр КОВАЛЬ

(підпис)

” ____ ” _____ 2021р.

ЗАВДАННЯ

на дипломну роботу студенту

Лесику Максиму Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Система моніторингу рівня глобальних ризиків

керівник роботи Караєва Наталія Веніамінівна, к.е.н., доц.

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

затверджена наказом вищого навчального закладу від ”24” травня 2021р. № **1167-с**

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи Python проект

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Проаналізувати існуючі програмні рішення та засоби аналізу рівня глобальних ризиків, спроектувати структуру бази даних спроектувати архітектуру програми, розробити програмне забезпечення, реалізувати інтерфейс користувача.

5. Перелік ілюстративного матеріалу

Структура системи, схожі системи, класифікація ризиків системи, концептуальна схема бази даних, діаграма прецедентів, вікно головного меню, вікна таблиць.

6. Консультанти розділів роботи

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

7. Дата видачі завдання ” 11 ” _____ жовтня _____ 2020 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітки
1.	Затвердження теми роботи	15.10.2020	
2.	Вивчення та аналіз задачі	16.10.2020- 16.12.2020	
3.	Розробка архітектури та загальної структури системи	09.03.2021- 23.03.2021	
4.	Розробка структур окремих підсистем	24.03.2020- 10.04.2020	
5.	Програмна реалізація системи	14.04.2021- 10.05.2021	
6.	Оформлення пояснювальної записки	18.05.2021- 02.06.2021	
7.	Захист програмного продукту	11.05.2021	
8.	Передзахист	25.05.2021	
9.	Захист	15.06.2021	

Студент _____

(підпис)

Лесик М.О. _____

(прізвище та ініціали)

Керівник роботи _____

(підпис)

Караєва Н.В. _____

(прізвище та ініціали)

АНОТАЦІЯ

У роботі розглянуті теоретичні та практичні аспекти розробки програмного продукту для моніторингу рівня глобальних ризиків.

Метою роботи є створення системи моніторингу рівня глобальних ризиків що є загальнодоступним ресурсом.

Для досягнення мети розроблено класифікаційну модель, реалізовано виводу даних та їх класифікації, та протестовано застосунок на реальних даних.

Пояснювальна записка складається зі вступу, п'яти розділів, висновку, списку використаних джерел; містить 49 сторінок, 17 рисунків, 5 таблиць та 4 додатки. Список використаних джерел включає 21 бібліографічних найменувань.

Ключові слова: система моніторингу, глобальні ризики, класифікація, база даних, засоби розробки.

ABSTRACT

The paper considers theoretical and practical aspects of software product development for monitoring the level of global risks.

The aim of the work is to create a system for monitoring the level of global risks, which is a publicly available resource.

To achieve this goal, a classification model was developed, data output and classification were implemented, and the application was tested on real data.

The explanatory note consists of an introduction, five sections, a conclusion, a list of sources used; contains 49 pages, 17 figures, 5 tables and 4 appendices. The list of used sources includes 21 bibliographic names.

Key words: monitoring system, global risks, classification, database, development tools.

ЗМІСТ

Перелік умовних позначень скорочень та термінів.....	9
Вступ.....	10
1 Постановка задачі проектування системи моніторингу рівня глобальних ризиків	12
2 Опис предметної області системи моніторингу рівня глобальних ризиків .	14
2.1 Світова практика дослідження глобальних ризиків.....	14
2.2 Моніторинг глобальних ризиків.....	16
3 Засоби розробки	21
3.1 Мова програмування Python	21
3.2 Бібліотека Tkinter	24
3.3 Система керування базами даних MySQL.....	25
3.4 Серведа MySQL Workbench.....	28
3.5 Серведа Jupyter Notebook	32
4 Опис програмної реалізації	35
4.1 Структурна схема.....	35
4.2 Концептуальна схема бази даних	36
4.3 Опис таблиць БД.....	38
5 Робота користувача з програмною системою	41
5.1 Опис системних вимог.....	41
5.2 Робота користувача з продуктом.....	41
Висновки	47
Список використаних джерел	48
Додаток А. Специфікація	50
Додаток Б. Текст програми	52
Додаток В. Опис програмного модулю	68
Додаток Г. Апробація	78

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

СКБД — система керування базами даних.

API (англ. Application Programming Interface) — прикладний програмний інтерфейс.

БД — база даних.

CRUD (англ. create read update delete) — 4 базові функції управління даними «створення, зчитування, зміна і видалення».

ORM — (англ. Object-relational mapping, Об'єктно-реляційна проекція) — технологія програмування, яка зв'язує бази даних з концепціями об'єктно-орієнтованих мов програмування.

SQL (англ. Select query language) — декларативна мова програмування для взаємодії користувача з базами даних.

WEF(ВЕФ) — Світовий економічний форум.

ВВП — валовий внутрішній продукт.

ВСТУП

Одним з найбільш тривожних і парадоксальних наслідків глобалізації є її вплив на ризики. Експертами міжнародної організації Всесвітнього економічного форуму (ВЕФ) поняття «глобальний ризик» визначається як невизначена подія чи стан, які у разі їх виникнення можуть спричинити значний негативний вплив для кількох країн або галузей промисловості протягом наступних 10 років [1].

Специфічна властивість глобальних ризиків полягає у тому, що глобальні ризики формуються і аналізуються на глобальному рівні, а їх вплив проявляється на макроекономічному рівні конкретної країни, причому ступінь впливу одного і того ж глобального ризику на різні макроекономіки неоднакова. Крім того, в умовах інтенсифікації процесу глобалізації, формується тренд неухильного зростання кількості глобальних ризиків, ймовірності виникнення і ступеня їх впливу а, також, зростає взаємовплив і взаємодія ризиків, надаючи впливу системний характер. У таких умовах обмежуються можливості окремо взятої країни (держави) у поодиночці контролювати і управляти глобальними ризиками. Вищезазначені специфічні властивості висувають особливі вимоги до побудови систем моніторингу глобальних ризиків і формування бази даних, що дозволяють розробляти ефективні заходи щодо попередження та управління ризиками. Організація системи моніторингу глобальних ризиків і можливості трансформації їх в ризики сталого розвитку України є необхідною і актуальною задачею.

У 15-му виданні звіту про глобальні ризики (Global risk report) Всесвітнього економічного форуму аналізуються ризики від переломів у соціальних мережах, що проявляються через постійні та нові ризики для здоров'я людини, зростання безробіття, розширення цифрових розривів, розчарування молоді та геополітичну фрагментацію. Підприємства ризикують безладним струсом, який може виключити з ринків майбутнього великі

когорти працівників і компаній. Деградація навколишнього середовища, як і раніше, є екзистенційною загрозою для людства ризикує перетнутися з переломами в суспільному середовищі, щоб принести серйозні наслідки. Тим не менш, коли світ більше піддається ризику, уроки можуть бути зроблені для зміцнення реагування та стійкості.

Робота містить 5 розділів.

У першому розділі описується постановка задачі створення системи моніторингу рівня глобальних ризиків.

У другому розділі описуються підходи до вирішення проблеми створення системи моніторингу рівня глобальних ризиків.

У третьому розділі вказуються основні засоби розробки даної системи.

У четвертому розділі дано опис реалізованого програмного продукту та його архітектури.

У п'ятому розділі описано роботу користувача з програмною системою.

1. ПОСТАНОВКА ЗАДАЧІ ПРОЕКТУВАННЯ СИСТЕМИ МОНІТОРИНГУ РІВНЯ ГЛОБАЛЬНИХ РИЗИКІВ

Призначення програмної системи моніторингу полягає в оцінюванні динаміки загроз рівня глобальних ризиків для подальшого реагування і вироблення регіональної політики, щодо їх мінімізації. Для моніторингу використовується система відповідних довідникових даних.

В результаті розробки системи, користувачу буде надана можливість працювати та аналізувати різні довідникові дані, усіх регіонів за останні роки, прогнозувати варіанти і відбирати найбільш ефективні стратегії мінімізації впливу ризиків на відокремлення подальшого економічного зростання та соціального розвитку держави.

Розроблений додаток повинен забезпечувати наступні можливості:

- аналіз динаміки зміни ризиків за роками;
- розподілення найімовірніших ризиків за регіонами;
- візуалізація отриманих результатів у табличному вигляді для подальшого аналізу.
- можливість доповнювати базу даних (БД).

Вхідна інформація: довідникові показники з щорічного звіту Всесвітнього економічного форуму про глобальні ризики.

Для розробки системи була використана мова програмування Python. Розробка графічного інтерфейсу користувача відбувалась на основі бібліотеки Tkinter (рисунок 1.1).

Метою розробки системи моніторингу рівня глобальних ризиків є створення програмного продукту, що дасть змогу, на основі довідникових даних займатись моніторингом динаміки зміни ризиків за кожним роком. Це дасть можливість зважено та чітко проаналізувати ефективність роботи

діючих превентивних заходів на зменшення впливу глобальних ризиків на екологію та економіку.

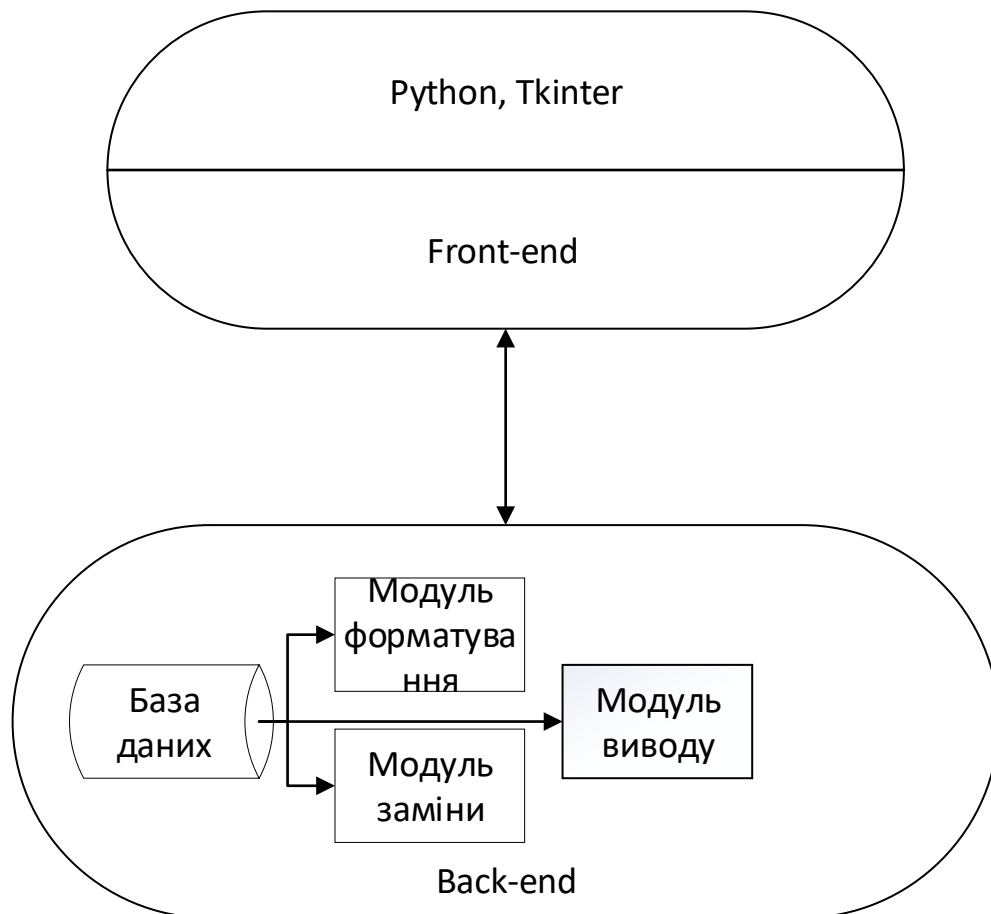


Рис. 1.1– Структура системи

Завдання цього проекту полягає в тому, щоб дати змогу звичайному користувачу ознайомитись з поняттям глобальний ризик та його впливу на галузі підприємства, також допомогти у визначенні стратегічних превентивних заходів та напрямів переходу економіки на траєкторію мінімізування впливу ризиків на засадах сталого розвитку відповідно до національних пріоритетів.

Програма надає користувачу такі функціональні можливості:

- продивитися і відредагувати базу даних статистичних показників;
- актуалізувати дані за нові періоди;
- видаляти дані з таблиць БД.

2. ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ СИСТЕМИ МОНІТОРИНГУ РІВНЯ ГЛОБАЛЬНИХ РИЗИКІВ

У розділі обґрунтовано нормативно-методичну основу формування системи моніторингу рівня глобальних ризиків. Визначено необхідність використання створення такої системи.

2.1 Світова практика дослідження глобальних ризиків

Всесвітній економічний форум (WEF), що базується в Кельні, Женевський кантон, Швейцарія [2], є міжнародною неурядовою організацією, заснованою 24 січня 1971 р. Місія ВЕФ зазначена як "завдання вдосконалення світу шляхом залучення бізнесу, політичних, наукових та інших лідерів для формування глобальних, регіональних та галузевих програм" (рисунок 2.1).



Рисунок 2.1– Сайт Всесвітнього економічного форуму

ВЕФ проводить щорічну зустріч наприкінці січня в Давосі, гірському курорті Граубюнден, у східних Альпах Швейцарії. На зустрічі збираються близько 3000 керівників підприємств, міжнародних політичних лідерів, економістів, знаменитостей та журналістів протягом п'яти днів для обговорення глобальних проблем протягом 500 сесій.

Організація також скликає близько шести — восьми регіональних зустрічей щороку в місцях по Африці, Східній Азії, Латинській Америці та Індії та проводить дві подальші щорічні зустрічі в Китаї та Об'єднаних Арабських Еміратах. Окрім зустрічей, організація надає платформу для лідерів усіх груп зацікавлених сторін з усього світу бізнесу, уряду та громадянського суспільства для співпраці у багатьох проектах та ініціативах. Він також випускає низку звітів та залучає своїх членів до галузевих ініціатив.

Фонд також діє як аналітичний центр, публікуючи широкий спектр звітів. Зокрема, "Команди стратегічного аналізу" зосереджуються на підготовці звітів, що мають відношення до конкурентоспроможності, глобальних ризиків та сценарію.

"Команда з питань конкурентоспроможності" видає цілий ряд щорічних економічних звітів (вперше опублікованих у дужках): Звіт про глобальну конкурентоспроможність (1979) вимірює конкурентоспроможність країн та економік; Глобальний звіт про інформаційні технології (2001) оцінював їх конкурентоспроможність на основі їхньої ІТ-готовності; у звіті про глобальні гендерні розриви розглядались найважливіші сфери нерівності між чоловіками та жінками; у звіті про глобальні ризики (2006 р.) оцінено ключові глобальні ризики; Глобальний звіт про подорожі та туризм (2007) вимірював конкурентоспроможність подорожей та туризму; Звіт про фінансовий розвиток (2008 р.) мав на меті надати країнам комплексні засоби для встановлення орієнтирів для різних аспектів їх фінансової системи та встановлення пріоритетів для вдосконалення; і Звіт про глобальну сприятливу торгівлю (2008 р.) представив міждержавний аналіз великої кількості заходів, що сприяють торгівлі між країнами [3].

"Мережа реагування на ризик" щорічно складає звіт з оцінкою ризиків, які, як вважають, входять до сфери дії цих команд, мають міжгалузеву актуальність, є невизначеними та можуть нанести економічну шкоду на 10 млрд. Доларів США, мають потенціал спричинити серйозні людські страждання, і які вимагають підходу з метою зменшення наслідків, який має багато зацікавлених сторін.

У 2020 році форум опублікував звіт під назвою: "Підвищення ризику для природи". У цій доповіді форум підрахував, що приблизно половина світового ВВП залежить сильно або помірно від природи, а 1 долар, витрачений на відновлення природи, дає 9 доларів прибутку.

2.1 Моніторинг глобальних ризиків

Проаналізувавши тенденції останніх років, представники Всесвітнього економічного форуму дійшли висновку, що саме екологічні ризики, а також ризики кіберсистем будуть все частіше потрапляти в розроблені рейтинги (таблиця 2.1). Крім того, що, незважаючи на те, що глобальні економічні ризики не потрапляли до рейтингу вже чотири роки, саме ці ризики та загроза економічній безпеці як окремих підприємств, так і держав у цілому, викликають велике занепокоєння світових експертів. Респонденти Всесвітнього економічного форуму відзначили свою стурбованість погіршенням міжнародної економічної кон'юнктури: переважна їх більшість очікує погіршення стану економічної безпеки, що пов'язане з економічними протистояннями основних держав (91%) та стиранням багатосторонніх торгових правил та угод (88%). Занепокоєння економічною безпекою та проблемою глобальних економічних ризиків висловлюють і національні економісти [1].

Таблиця 2.1 – Топ-5 глобальних ризиків за різними критеріями

Рік	Місце ризику				
	1	2	3	4	5
<i>за ступенем вірогідності прояву</i>					
2018	екстремальні погодні явища ²	значні природні катастрофи ²	кібер-атаки ⁵	масові випадки крадіжки даних/ шахрайства ⁵	провали заходів пом'якшення наслідків зміни клімату ²
2019	екстремальні погодні явища ²	провали заходів пом'якшення наслідків зміни клімату ²	значні природні катастрофи ²	масові випадки крадіжки даних/ шахрайства ⁵	кібер-атаки ⁵
2020	екстремальні погодні явища ²	провали заходів пом'якшення наслідків зміни клімату ²	значні природні катастрофи ²	втрата біорізноманіття ²	антропогенно природні катастрофи ²
<i>з точки зору впливу події, якщо вона станеться</i>					
2018	зброя масового ураження ³	екстремальні погодні явища ²	природні катастрофи ²	провали заходів пом'якшення наслідків зміни клімату ²	водна криза (водозабезпечення) ⁴
2019	зброя масового ураження ³	провали заходів пом'якшення наслідків зміни клімату ²	екстремальні погодні явища ²	водна криза (водозабезпечення) ⁴	природні катастрофи ²
2020	провали заходів пом'якшення наслідків в зміни клімату ²	зброя масового ураження ³	втрата біорізноманіття ²	екстремальні погодні явища ²	водна криза (водозабезпечення) ⁴
Прим: ¹ – економічні; ² – екологічні; ³ – геополітичні; ⁴ – соціальні; ⁵ – технологічні					

Глобальні ризики, що представлені у таблиці 2.1, притаманні й Україні. Саме тому економісти в цілому мають зважати на них та розробляти

превентивні заходи ризик-менеджменту, серед яких: виявлення, оцінка, профілактика, страхування ризиків.

Географічна складова дослідження дозволяє сконцентрувати увагу на глобалізаційних ризиках, що притаманні окремому регіону світу .

Для Європи та України характерними є такі ризики [2]:

- безробіття або неповна зайнятість;
- значна міграція;
- глибока соціальна нестабільність;

Для Середньозахідної та Північної Африки:

- криза водопостачання;
- соціальна нестабільність;
- міждержавні конфлікти;

Для Південної Африки:

- безробіття або неповна зайнятість;
- продовольча криза;
- поширення інфекційних хвороб;

Для Північної Америки є характерними такі ризики:

- масштабні кібератаки;
- погіршення інфраструктури;
- складнощі адаптації до змін клімату;

Країни Латинської Америки з високою ймовірністю відчують:

- глибоку соціальну нестабільність;
- крах національного управління;

Для Центральної Азії:

- обвал цін на енергоносії;
- терористичні атаки;
- міждержавні конфлікти.

Для країн Східної Азії та Тихоокеанського регіону:

- міждержавні конфлікти;

- штучні екологічні катастрофи.

Для Південної Азії:

- терористичні атаки;
- криза водопостачання.

Основною метою дослідження є: виявлення та запобігання ризикам або підвищення стійкості проти них. Всі ризики об'єднані за п'ятьма групами: економічні, соціальні, геополітичні, екологічні та технологічні (рисунок 2.2).

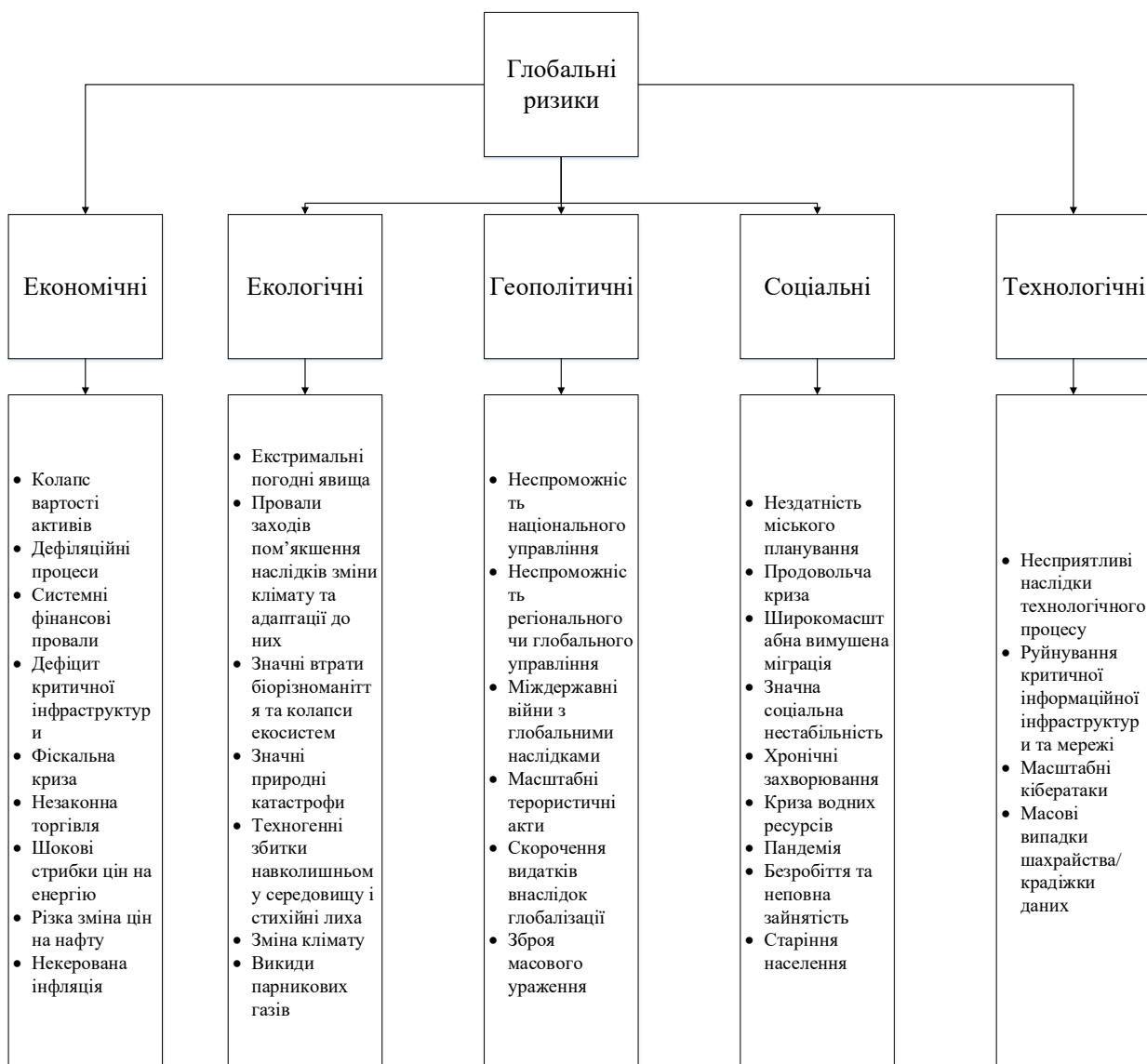


Рисунок 2.2 – Класифікація ризиків

Така класифікація дає змогу зручно аналізувати ризики при їх дослідженні, що в свою чергу дає змогу як найефективніше впроваджувати

превентивні заходи боротьби з ними. Що в свою чергу полегшує становище економіки певних країн в цілому.

Ризики досліджуються відповідно до ймовірності їх настання та сили впливу. За ймовірністю настання ризики характеризуються та досліджуються безпосередньо перед досліджуваним роком, що дає змогу легко підготуватися до них економістам та країнам яким ці ризики загрожують. За силою впливу ризики характеризуються як висновок пройденого року, та аналізуються які з них мали найбільший вплив на економіку світу, що в свою чергу дає змогу зрозуміти чи будуть вони небезпечними для поточного року.

3. ЗАСОБИ РОЗРОБКИ

При проектуванні застосунку важливим є правильний підхід до вибору засобів розробки, що впливає на час та якість розробки в цілому. Також це впливає на надійність та швидкість продукту. Та не слід забувати про особливості операційної системи для якої проектується цей додаток.

3.1 Мова програмування Python

Мова програмування Python (в українській мові зустрічаються назви пітон або пайтон) — високорівнева мова програмування загального призначення з динамічною строгою типізацією і автоматичним управлінням пам'яттю, орієнтований на підвищення продуктивності розробника, читання коду і його якості, а також на забезпечення переносимості написаних на ньому програм. Мова є повністю об'єктно-орієнтованою — все є об'єктами. Незвичайною особливістю мови є виділення блоків коду пробільними відступами. Синтаксис ядра мови мінімалістичний, за рахунок чого на практиці рідко виникає необхідність звертатися до документації. Сама ж мова як інтерпретується так і використовується в тому числі для написання скриптів. Недоліками мови є часто більш низька швидкість роботи і більш високе споживання пам'яті написаних на ньому програм в порівнянні з аналогічним кодом, написаним на компільованих мовах, таких як C або C ++ [4].

Мова програмування Python є мультипарадигмальною, що підтримує імперативне, процедурне, структурне, об'єктно-орієнтоване програмування, метапрограмування і функціональне програмування. Завдання узагальненого програмування вирішуються за рахунок динамічної типізації. Аспектно-орієнтоване програмування частково підтримується через декоратори, більш повноцінна підтримка забезпечується додатковими фреймворками. Такі методики як контрактне і логічне програмування можна реалізувати за

допомогою бібліотек або розширень. Основні архітектурні риси — динамічна типізація, автоматичне керування пам'яттю, повна інтроспекція, механізм обробки виключень, підтримка багатопоточних обчислень з глобальної блокуванням інтерпретатора (GIL), високорівневі структури даних. Підтримується розбиття програм на модулі, які, в свою чергу, можуть об'єднуватися в пакети [5].

Еталонною реалізацією Python є інтерпретатор CPython, що підтримує більшість активно використовуваних платформ і є стандартом мови. Він поширюється під вільною ліцензією Python Software Foundation License, що дозволяє використовувати його без обмежень в будь-яких додатках, включаючи пропрієтарні. CPython компілює вихідні тексти в високорівневий байт-код, який виконується в стековій віртуальній машині. До інших трьох основних реалізацій мови відносяться Jython (для JVM), IronPython (для CLR / .NET) і PyPy. PyPy написаний на підмножині мови Python (RPython) і розроблявся як альтернатива CPython з метою підвищення швидкості виконання програм, в тому числі за рахунок використання JIT-компіляції. Підтримка версії Python 2 закінчилася в 2020 році. На поточний момент активно розвивається версія мови Python 3. Розробка мови ведеться через пропозиції щодо розширення мови PEP (англ. Python Enhancement Proposal), в яких описуються нововведення, робляться коригування відповідно до зворотного зв'язку від спільноти і документуються підсумкові рішення [6].

Стандартна бібліотека включає великий набір корисних функцій, починаючи від функціоналу для роботи з текстом і закінчуючи засобами для написання мережевих додатків. Додаткові можливості, такі як математичне моделювання, робота з обладнанням, написання веб-додатків або розробка ігор, можуть реалізовуватися за допомогою великої кількості сторонніх бібліотек, а також інтеграцією бібліотек, написаних на C або C ++, при цьому і сам інтерпретатор Python може інтегруватися в проекти, написані на цих мовах. Існує і спеціалізований репозиторій програмного забезпечення, написаного на Python, PyPI. Даний репозиторій надає кошти для простої установки пакетів в

операційну систему і став стандартом для Python. Станом на 2019 рік у ньому містилося понад 175 тисяч пакетів [7].

Мова програмування Python стала однією з найпопулярніших мов, вона використовується в аналізі даних, машинному навчанні, DevOps і веб-розробки, а також в інших сферах, включаючи розробку ігор. За рахунок читабельності, простого синтаксису і відсутності необхідності в компіляції мова добре підходить для навчання програмуванню, дозволяючи концентруватися на вивченні алгоритмів, концептів і парадигм. Налагодження ж і експериментування в значній мірі полегшуються тим фактом, що мова інтерпретується. Застосовується мова багатьма великими компаніями, такими як Google або Facebook. Станом на квітень 2021 року Python займає третє місце в рейтингу TIOBE популярності мов програмування з показником 11,03%. «Мовою року» за версією TIOBE Python оголошувався в 2007, 2010, 2018 і 2020 году [8].

Мова програмування Python портована і працює майже на всіх відомих платформах від КПК до мейнфреймів. Існують порти під Microsoft Windows, практично під всі варіанти UNIX (включаючи FreeBSD і Linux), Android, Plan 9, і т.д [9].

У міру старіння платформи її підтримка в основній гілці мови припиняється. Наприклад, з версії 2.6 припинена підтримка Windows 95, Windows 98 і Windows ME. У версії 3.5 перестала підтримуватися Windows XP. У версії 3.9 перестала підтримуватися Windows Vista і Windows 7.

При цьому, на відміну від багатьох портованих систем, для всіх основних платформ Python має підтримку характерних для даної платформи технологій (наприклад, Microsoft COM / DCOM). Більш того, існує спеціальна версія Python для віртуальної машини Java — Jython, що дозволяє інтерпретатору виконуватися на будь-якій системі, що підтримує Java, при цьому класи Java можуть безпосередньо використовуватися з Python і навіть бути написаними на Python. Також кілька проектів забезпечують інтеграцію з платформою Microsoft.NET, основні з яких IronPython і Python.Net [10].

Мова програмування Python підтримує динамічну типізацію, тобто тип змінної визначається лише під час виконання. Тому замість «присвоювання значення змінної» краще говорити про «зв'язуванні значення з деяким ім'ям». До примітивних типів в Python відносяться булевий, ціле число довільної точності, число з плаваючою комою і комплексне число. З контейнерних типів в Python вбудовані: рядок, список, кортеж, словник. Всі значення є об'єктами, в тому числі функції, методи, модулі, класи.

Додати новий тип можна або написавши клас (class), або визначивши новий тип в модулі розширення (наприклад, написаному на мові C). Система класів підтримує спадкування (одиначне і множинне) і метапрограмування. Можливо успадкування від більшості вбудованих типів і типів розширень.

Мова програмування Python є мультипарадигмальною. Повністю підтримуються об'єктно-орієнтоване, структурний, узагальнена, функціональне програмування і метапрограмування. Базова підтримка аспектно-орієнтованого програмування реалізується за рахунок метапрограмування. Безліч інших методик, в тому числі контрактне і логічне програмування можна реалізувати за допомогою розширень [11].

3.2 Бібліотека Tkinter

Бібліотека Tkinter (від англ. Tk interface) - крос-платформна подієво-орієнтована графічна бібліотека на основі засобів Tk (широко поширена в світі GNU / Linux та інших UNIX-подібних систем, перенесена також і на Microsoft Windows), написана Стін Лумхольтом (Steen Lumholt) і Гвідо ван Россум. Входить в стандартну бібліотеку Python [12].

Бібліотека Tkinter — це вільне програмне забезпечення, яке розповсюджується під Python-ліцензією.

Бібліотека Tkinter — це пакет для Python, призначений для роботи з бібліотекою Tk. Бібліотека Tk містить компоненти графічного інтерфейсу

користувача (graphical user interface — GUI). Ця бібліотека написана на мові програмування Tcl.

Під графічним інтерфейсом користувача (GUI) маються на увазі всі ті вікна, кнопки, текстові поля для введення, скролери, списки, радіокнопки, прапорці та ін., Які ви бачите на екрані, відкриваючи ту чи іншу програму. Через них ви взаємодієте з програмою і керуєте нею. Всі ці елементи інтерфейсу будемо називати віджетами (widgets).

В даний час майже всі програми, які створюються для кінцевого користувача, мають GUI. Рідко які програми, які передбачають взаємодію з людиною, залишаються консольними.

Програми з графічним інтерфейсом користувача подієво-орієнтовані. Ви вже повинні мати уявлення про структурний і бажано об'єктно-орієнтованому програмуванні. Подієво-орієнтоване орієнтоване на події. Тобто та чи інша частина програмного коду починає виконуватися лише тоді, коли трапляється ту чи іншу подію.

Подієво-орієнтоване програмування базується на об'єктно-орієнтованому і структурному. Навіть якщо ми не створюємо власних класів та об'єктів, то все-одно ними користуємося. Всі віджети — об'єкти, породжені вбудованими класами.

Події бувають різними. Спрацював часовий чинник, хтось клікнув мишкою або натиснув Enter, почав вводити текст, перемкнув радіокнопки, прокрутив сторінку вниз і т. Д. Коли трапляється щось подібне, то, якщо був створений відповідний обробник, відбувається спрацьовування певної частини програми, що приводить до якого-небудь результату.

3.3 Система керування базами даних MySQL

Реляційна база даних — це набір даних з зумовленими зв'язками між ними. Ці дані організовані у вигляді набору таблиць, що складаються із

стовпців і рядків. У таблицях зберігається інформація про об'єкти, представлених в базі даних. У кожному стовпчику таблиці зберігається певний тип даних, в кожному осередку — значення атрибута. Кожен стовпчик таблиці являє собою набір пов'язаних значень, що відносяться до одного об'єкту або сутності. Кожен рядок в таблиці може бути позначений унікальним ідентифікатором, званим первинним ключем, а рядки з декількох таблиць можуть бути пов'язані з допомогою зовнішніх ключів. До цих даних можна отримати доступ багатьма способами, і при цьому реорганізувати таблиці БД не потрібно [13].

СКБД MySQL вільна реляційна система управління базами даних. Розробка та підтримка сайту MySQL здійснює корпорація Oracle, яка отримала права на торговельну марку разом з поглиненої Sun Microsystems, яка раніше придбала шведську компанію MySQL AB. Продукт поширюється як під GNU General Public License, так і під власною комерційною ліцензією. Крім цього, розробники створюють функціональність за замовленням ліцензійних користувачів. Саме завдяки такому замовленню в найраніших версіях з'явився механізм реплікації [14].

СКБД MySQL є рішенням для малих і середніх додатків. Входить до складу серверів WAMP, AppServ, LAMP і в портативні збірки серверів Денвер, XAMPP, VertrigoServ [15]. Зазвичай MySQL використовується як сервер, до якого звертаються локальні або віддалені клієнти, проте в дистрибутив входить бібліотека внутрішнього сервера, що дозволяє включати MySQL в автономні програми.

Гнучкість СКБД MySQL забезпечується підтримкою великої кількості типів таблиць: користувачі можуть вибрати як таблиці типу MyISAM, що підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів. Більш того, СУБД MySQL поставляється із спеціальним типом таблиць EXAMPLE, що демонструє принципи створення нових типів таблиць. Завдяки відкритій архітектурі і GPL-ліцензуванню, в СКБД MySQL постійно з'являються нові типи таблиць [16].

26 лютого 2008 року Sun Microsystems придбала MySQL AB за 1 млрд доларів, 27 січня 2010 року Oracle придбала Sun Microsystems за 7,4 млрд доларів і включила MySQL в свою лінійку СУБД.

Спільнотою розробників MySQL створені різні відгалуження коду, такі як Drizzle (англ.), OurDelta, Percona Server і MariaDB. Всі ці відгалуження вже існували на момент поглинання компанії Sun корпорацією Oracle.

СКБД MySQL виникла як спроба застосувати mSQL до власних розробок компанії: таблицям, для яких використовувалися ISAM підпрограми низького рівня. В результаті був вироблений новий SQL-інтерфейс, але API-інтерфейс залишився в спадок від mSQL. Звідки походить назва «MySQL» достеменно невідомо. Розробники дають два варіанти: або тому, що практично всі напрацювання компанії починалися з префікса «Му ...» (з англ. - «моє ...»), або на честь дочки Мю (фін. My) одного з розробників системи Мікаеля Віденіус [17].

Логотип MySQL у вигляді дельфіна носить ім'я «Sakila». Він був вибраний з великого списку запропонованих користувачами «імен дельфіна». Ім'я «Sakila» було відправлено Open Source-розробником Ambrose Twebaze.

СКБД MySQL має подвійне ліцензування. MySQL може поширюватися відповідно до умов ліцензії GPL. Однак за умовами GPL, якщо яка-небудь програма використовує бібліотеки (або включає в себе інший GPL-код) MySQL, то вона теж повинна розповсюджуватися за ліцензією GPL. Це може розходитися з планами розробників, які не бажають відкривати вихідні тексти своїх програм. Для таких випадків передбачена комерційна ліцензія, яка також забезпечує якісну сервісну підтримку. Для вільного програмного забезпечення Oracle надає окреме виключення з правил, явно дозволяє використання і поширення MySQL разом з ПО, поширюваним під ліцензією з певного Oracle списку [18].

СКБД MySQL має API та конектори для мов Delphi, C, C ++, Ейфель, Java, Лісп, Perl, PHP, Python, Ruby, Smalltalk, Pascal, Tcl і Lua, бібліотеки для

мов платформи .NET, а також забезпечує підтримку для ODBC за допомогою ODBC-драйвера MyODBC.

Конектор MyODBC є драйвер ODBC (2.50) рівня 0 (з деякими можливостями рівнів 1 і 2) для під'єднання сумісного з ODBC додатки до MySQL. MyODBC працює на всіх системах Microsoft Windows і на більшості платформ Unix.

Максимальний розмір таблиць в MySQL 3.22 до 4 гігабайт, в наступних версіях обмежень немає [19].

Розмір таблиці обмежений її типом. У загальному випадку тип MyISAM обмежений граничним розміром файлу в файлової системі операційної системи. Наприклад, в NTFS цей розмір теоретично може бути до 32 ексабайт. У разі InnoDB одна таблиця може зберігатися в декількох файлах, які мають єдине табличний простір. Розмір останнього може досягати 64 терабайт.

На відміну від MyISAM, в InnoDB є значне обмеження на кількість стовпців, яке можна додати в одну таблицю. Розмір сторінки пам'яті за замовчуванням становить 16 кілобайт, з яких під дані відведено 8123 байта. Розмір покажчика на динамічні поля становить 20 байт. Таким чином, в разі використання динамічного формату рядка (`ROW_FORMAT = DYNAMIC`), одна таблиця може вмістити максимум 409 стовпців типу blob або text [20].

3.4 Середа MySQL Workbench

Середа MySQL Workbench - це уніфікований візуальний інструмент для архітекторів баз даних, розробників та DBA. MySQL Workbench забезпечує моделювання даних, розробку SQL та комплексні інструменти адміністрування для конфігурації сервера, введення користувача, резервне копіювання та багато іншого. MySQL Workbench доступний у Windows, Linux та Mac OS X.

Середа MySQL Workbench дозволяє DBA, розробнику або архітектору даних візуально розробляти, моделювати, генерувати та керувати базами

даних. Вона включає в себе все, що потребує для моделей даних та створення складних моделей ER, Forward та Reverse Engineering, а також доставляє ключові функції для виконання складних змін та завдань документації, які зазвичай вимагають багато часу та зусиль.

Середа MySQL Workbench спрощує проектування та обслуговування баз даних, автоматизує трудомісткі та схильні до помилок завдання, а також покращує зв'язок між DBA та командами розробників. Це дозволяє архітекторам даних візуалізувати вимоги, спілкуватися із зацікавленими сторонами та вирішувати питання проектування до того, як буде зроблено великі інвестиції часу та ресурсів. Це дозволяє керувати моделями баз даних, що є найефективнішою методологією для створення діючих та якісно працюючих баз даних, забезпечуючи при цьому гнучкість реагування на зміни, що змінюються у бізнесі. Службові програми перевірки моделей та схем забезпечують застосування найкращих практичних стандартів моделювання даних, а також застосовують специфічні для MySQL стандарти фізичного проектування, щоб не допускати помилок при створенні нових діаграм ER або генерації фізичних баз даних MySQL.

Середа MySQL Workbench надає можливості для прямого проектування фізичних конструкцій баз даних. Візуальну модель даних можна легко перетворити у фізичну базу даних на цільовому сервері MySQL лише за кілька кліків миші. Весь код SQL автоматично генерується і запускається з першого разу, що виключає звичайний процес схильності до помилок ручного написання складного коду SQL. MySQL Workbench також дає змогу здійснити реверсивне проектування існуючої бази даних або упакованого додатка, щоб отримати кращу уяву про дизайн бази даних. MySQL Workbench не тільки може переадресувати та зворотно проектувати існуючі бази даних, але також може імпортувати сценарії SQL для побудови моделей та експортувати моделі до сценаріїв DDL, які можна запускати пізніше.

Управління змінами в базі даних - це складний і складний процес, який передбачає підтримку різних версій схем баз даних та ручну модифікацію

існуючих баз даних. Щоб допомогти адміністраторам баз даних та розробникам в управлінні змінами, MySQL Workbench включає утиліти синхронізації та порівняння схем. DBA може порівнювати дві реальні бази даних або модель та реальну базу даних і візуально бачити відмінності, а також виконувати синхронізацію між моделлю та реальною базою даних або навпаки.

Документування дизайну бази даних може бути трудомістким процесом. MySQL Workbench включає DBDoc, який дозволяє базі даних або розробнику надавати документацію бази даних, що вказує і клацає. Моделі можуть бути задокументовані як у форматі HTML, так і у форматі звичайного тексту, і вони включають усі об'єкти та моделі в поточному сеансі MySQL Workbench.

Середа MySQL Workbench забезпечує візуальні інструменти для створення, виконання та оптимізації запитів SQL. Редактор SQL забезпечує підсвічування кольорового синтаксису, автоматичне завершення, повторне використання SQL-фрагментів, а також історія виконання SQL. Панель підключення баз даних дозволяє розробникам легко керувати стандартними підключеннями баз даних, включаючи MySQL. Інтернет-браузер забезпечує миттєвий доступ до схеми та об'єктів бази даних.

Середа MySQL Workbench забезпечує візуальну консоль, щоб легко керувати середовищами MySQL та отримати кращу видимість в бази даних. Розробники та DBAS можуть використовувати візуальні інструменти для налаштування серверів, керуючи користувачів, виконуючи резервну копію та відновлення, перевіряючи дані аудиту та перегляду здоров'я бази даних.

Середа MySQL Workbench надає набір інструментів для підвищення продуктивності програм MySQL. DBAS може швидко переглядати ключові показники ефективності, використовуючи інформаційну панель продуктивності. Звіти про ефективність забезпечують легку ідентифікацію та доступ до точок доступу до ІО, високі витрати SQL-заяви тощо. Крім того, з 1 натиском, розробники можуть побачити, де оптимізувати свій запит з покращеним та простим у використанні плану Visual Explain.

Сервєра MySQL Workbench тепер забезпечує повну, просту у використанні рїшення для перемїщення Microsoft SQL Server, Microsoft Access, Sybase ASE, PostgreSQL та інші таблиці RDBMS, об'єкти та дані до MySQL. Розробники та DBAS можуть швидко та легко перетворити існуючі програми, щоб запусити на MySQL як на Windows, так і на інші платформи.

Візуальний редактор SQL дозволяє розробникам створювати, редагувати та запускати запити, створювати та редагувати дані, а також переглядати та експортувати результати. Колірне підсвічування синтаксису, контекстна довідка та автоматичне заповнення допоміжних засобів при написанні та налагодженні операторів SQL. А плани ПОЯСНЕННЯ можна легко зібрати, щоб допомогти оптимізувати запити. Редактор SQL забезпечує:

- Редактор таблиць - легко редагувати дані та фіксувати зміни, використовуючи простий формат сітки.
- Вікно результатів - виконувати кілька запитів одночасно та переглядати текстові результати, зображення та просторові подання, а також статистику ефективності запитів на окремих вкладках.
- Панель фрагментів SQL - зберігайте та легко використовуйте загальний код Selects, DML та DDL.
- Панель історії - переглянути повну історію сеансів запитів та операторів, що показують, які запити виконувались і коли. Легко отримувати, переглядати, повторно запускати, додавати або змінювати раніше виконані оператори SQL.
- Панель пошуку даних таблиці - пошук даних у всій базі даних, довівши пошук тексту за будь-якою кількістю таблиць та схем, показуючи рядки, що відповідають заданому шаблону.
- Експортувати результати - експортувати дані результатів до загальних форматів, включаючи CSV, HTML та XML.

3.5 Середи Jupyter Notebook

Jupyter Notebook (колишні ноутбуки IPYTHON) — це веб-інтерактивне обчислювальне середовище для створення документів ноутбуків Jupyter. Термін "Notebook" може розробляти поводження з багатьма різними об'єктами, головним чином, веб-програма Jupyter, веб-сервера Jupyter Python або формат документа Jupyter, залежно від контексту. Документ Jupyter Notebook — це документ JSON, дотримуючись версії схеми, що містить упорядкований список елементів вводу / виводу, який може містити код, текст, як правило, закінчується розширенням ".ipynb" .

Jupyter Notebook може бути перетворений на ряд відкритих стандартних форматів виводу через "Завантажити як" у веб-інтерфейсі через бібліотеку NBConvert або "Jupyter" NBConvert " інтерфейс командного рядка в оболонці. Щоб спростити візуалізацію документів Jupyter Notebook в Інтернеті, бібліотека NBConvert надається як служба через NBViewer, який може взяти URL-адресу до будь-якого загальнодоступного ноутбука, конвертувати його на HTML на льоту та відображати його користувач. Jupyter Notebook забезпечує використання браузера, побудований на ряді популярних бібліотек з відкритим кодом:

- IPythonNotebook
- TornadoR
- jQuery
- Wordpress
- MathJax.

Jupyter Notebook може підключитися до багатьох ядер, що дозволяє програмування на різних мовах. За замовчуванням Jupyter Notebook з ядром IPYTHON.

Ядро Jupyter – це програма, відповідальна за обробку різних видів запитів (виконання коду, завершення коду, перевірка) та надання відповіді.

Ядра розмовляють з іншими компонентами Jupyter за допомогою Zerotmq, і, таким чином, можуть бути на тих самих або віддалених машинах. На відміну від багатьох інших ноутбуків-подібних інтерфейсів, у Jupyter, ядра не знають, що вони прикріплені до конкретного документа, і можуть бути пов'язані з багатьма клієнтами відразу. Зазвичай ядра дозволяють виконувати лише одну мову, але є пара винятків.

Хаб JupyterHub — багатокористувацький сервер для ноутбуків Jupyter. Він призначений для підтримки багатьох користувачів шляхом створення, управління та проксі-сервера багатьох окремих серверів Jupyter Notebook. Хоча JupyterHub вимагає управління серверами, сторонні служби, такі як Juroo, забезпечують альтернативу JupyterHub, розміщуючи та керуючи мульти- користувач блокноти Jupyter у хмарі.

Інтерфейс JupyterLab — це новіший користувацький інтерфейс для проекту Jupyter. Він пропонує будівельні блоки класичного блокнота Jupyter (блокнот, термінал, текстовий редактор, браузер файлів, розширені виходи тощо) в гнучкому користувацькому інтерфейсі. Перший стабільний реліз був оголошений 20 лютого 2018 р.

Система Jupyter Book — це проект з відкритим кодом для створення книг та документів з обчислювальних матеріалів. Це дозволяє користувачеві створювати вміст у суміші Markdown, розширеної версії Markdown під назвою MyST, Maths & Equations за допомогою MathJax, Jupyter Notebooks, reStructuredText, вихідних даних запущених Jupyter Notebooks під час побудови. Можна створити декілька вихідних форматів (на даний момент окремі файли, багатосторінкові веб-сторінки HTML та файли PDF).

nbgrader — це інструмент для створення та оцінки (маркування) завдань у блокнотах Jupyter. Це дозволяє інструктору створювати завдання, які включають вправи на кодування в python або будь-які інші підтримувані відповіді ядра та тексту. Подані завдання можуть бути автоматично позначені, оцінені вручну або суміш обох.

Jupyter Notebook став популярним користувальницьким інтерфейсом для хмарних обчислень, і основні хмарні провайдери прийняли Jupyter Notebook або похідні інструменти як інтерфейс для хмарних користувачів. Прикладами є ноутбуки SageMaker від Amazon, Colaboratory від Google та ноутбук Azure від Microsoft.

Google Colaboratory (також відомий як Colab) - це безкоштовне середовище для ноутбуків Jupyter, яке працює в хмарі та зберігає свої ноутбуки на Google Drive. Спочатку Colab був внутрішнім проектом Google; була зроблена спроба відкрити весь код і працювати більш безпосередньо вгору за течією, що призвело до розвитку розширення Google Chrome "Відкрити в Colab", але це врешті закінчилося, і розвиток Colab продовжився внутрішньо. Починаючи з жовтня 2019 року, інтерфейс Colaboratory дозволяє створювати лише блокноти з ядрами Python 2 та Python 3; однак, існуючий блокнот, ядро якого має IR або Swift, також буде працювати, оскільки в контейнері встановлені як R, так і Swift. Мова Джулії також може працювати на Colab (наприклад, з Python і графічними процесорами; модулі обробки тензорів Google також працюють з Джулією на Colab).

4. ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

4.1 Структурна схема

Діаграма варіантів використання (англ. Use case diagram) в UML — діаграма, що відображає відносини між акторами і прецедентами і є складовою частиною моделі прецедентів, що дозволяє описати систему на концептуальному рівні.

Прецедент — можливість модельованої системи (частина її функціональності), завдяки якій користувач може отримати конкретний, вимірний і потрібний йому результат. Прецедент відповідає окремому сервісу системи, визначає один з варіантів її використання і описує типовий спосіб взаємодії користувача з системою. Варіанти використання зазвичай застосовуються для специфікації зовнішніх вимог до системи.

Основне призначення діаграми — опис функціональності і поведінки, що дозволяє замовнику, кінцевому користувачеві і розробнику спільно обговорювати проєктовану або існуючу систему.

При моделюванні системи за допомогою діаграми прецедентів системний аналітик прагне:

- чітко відокремити систему від її оточення;
- визначити дійових осіб (акторів), їх взаємодія з системою і очікувану функціональність системи;
- визначити в глосарії предметної області поняття, які стосуються детального опису функціональності системи (тобто прецедентів).

Робота над діаграмою може початися з текстового опису, отриманого при роботі із замовником. При цьому нефункціональні вимоги (наприклад, конкретний мову або система програмування) при складанні моделі прецедентів опускаються (для них складається інший документ).

Діаграма прецедентів для системи зображено на рисунку 4.1.

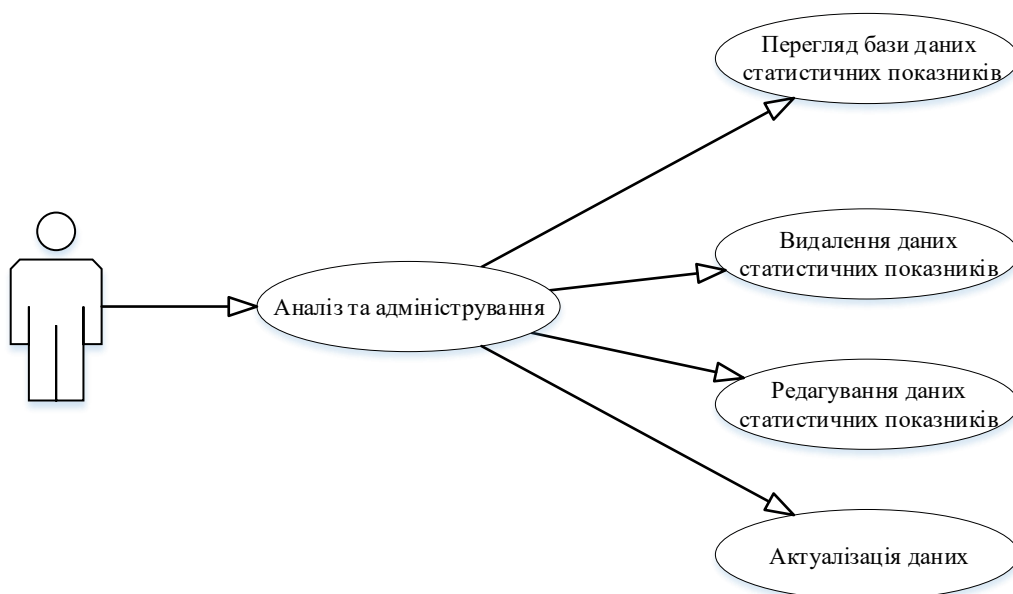


Рисунок 4.1 – Діаграма прецедентів

4.2 Концептуальна схема бази даних

Концептуальна модель — це модель, представлена безліччю понять і зв'язків між ними, що визначають смислову структуру розглянутої предметної області або її конкретного об'єкта.

Концептуальна модель — модель предметної області, що складається з переліку взаємопов'язаних понять, які використовуються для опису цієї області, разом з властивостями і характеристиками, класифікацією цих понять, за типами, ситуацій, ознаками в даній області і законів протікання процесів в ній. (Тлумачний словник по штучному інтелекту)

Концептуальна (змістовна) модель — це структура, що моделюється, властивості її елементів і причинно-наслідкові зв'язки, властиві системі і суттєві для досягнення мети моделювання.

Концептуальна модель відноситься до моделей, які формуються після процесу концептуалізації або узагальнення. Концептуальні моделі часто є абстракціями речей в реальному світі. Для різних етапів формування концепту

актуальні семантичні дослідження. Семантика — це поняття, значення, яке мислячі істоти надають різним елементам свого досвіду.

Основне завдання концептуальної моделі — передати фундаментальні принципи і основні функціональні можливості системи, яку вона представляє. Крім того, концептуальна модель повинна бути розроблена таким чином, щоб забезпечити легко зрозумілу системну інтерпретацію для користувачів моделі. Концептуальна модель, якщо вона реалізується належним чином, повинна задовольняти чотирьом основним цілям.

- Поліпшення розуміння індивідом репрезентативною системи
- Сприяння ефективній передачі деталей системи між зацікавленими сторонами
- Надання розробникам систем точки відліку для отримання специфікацій системи
- Документація системи для подальшого використання і забезпечення засобів для співпраці

Концептуальна модель грає важливу роль в загальному життєвому циклі розвитку системи. Очевидно, що якщо концептуальна модель не буде повністю розроблена, то реалізація фундаментальних властивостей системи може не бути реалізована належним чином, що призведе до майбутніх проблем або недоліків системи. Ці збої дійсно відбуваються в галузі і були пов'язані з відсутністю призначеного для користувача введення, неповними, неясними або мінливими вимогами. Ці слабкі ланки в процесі проектування і розробки системи можна простежити до неправильного виконання фундаментальних завдань концептуального моделювання. Важливість концептуального моделювання очевидна, коли такі системні збої пом'якшуються ретельною розробкою системи і дотриманням перевірених цілей та методів розвитку.

На рисунку 4.2 наведено концептуальну схему системи.

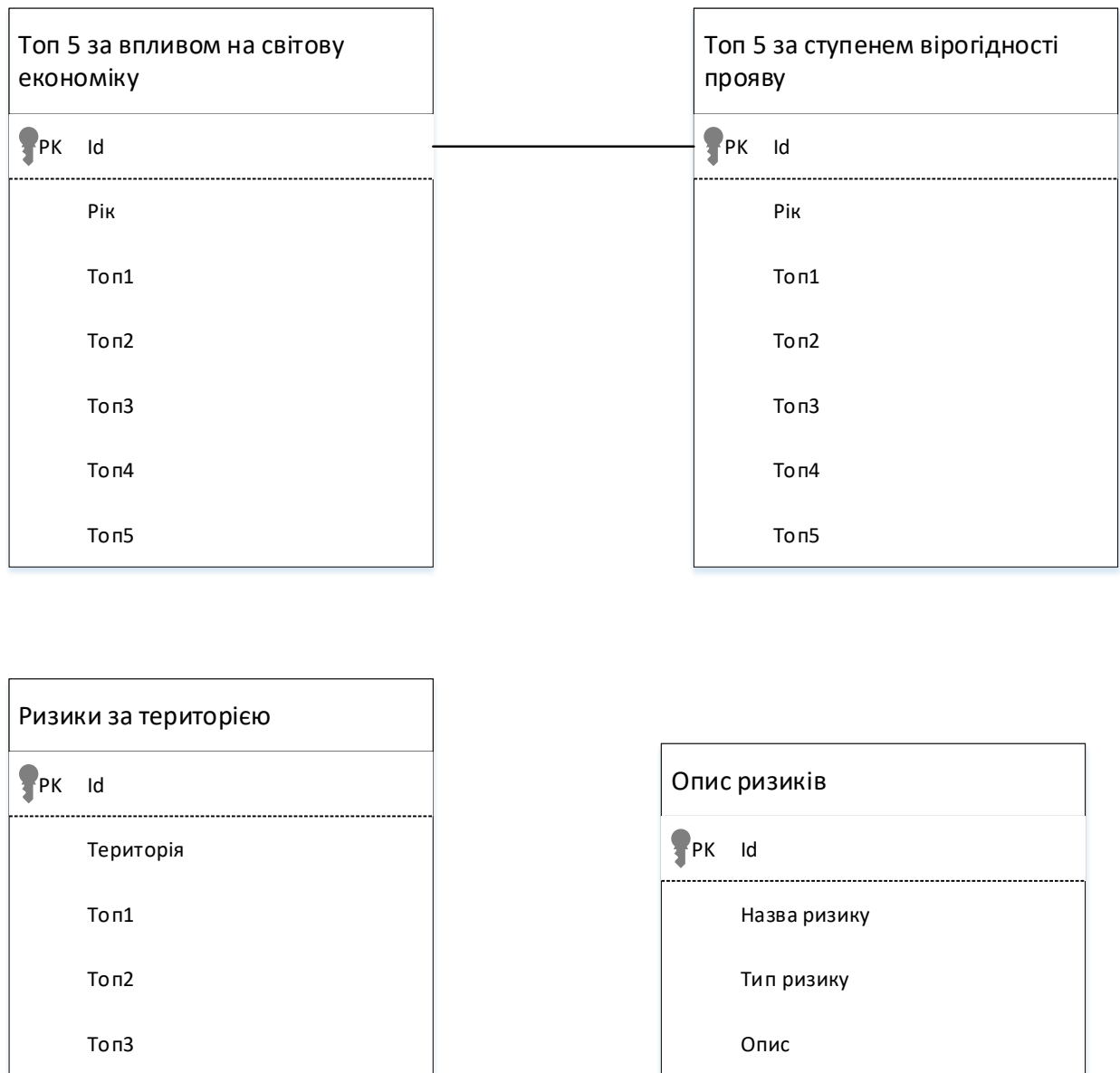


Рисунок 4.2 – Концептуальна схема бази даних

4.3 Опис таблиць БД

Для доступу до бази даних використовуємо MySQL Workbench. Для кожної таблиці створюється клас, який буде вміщати в себе властивості та відображати БД у вигляді об'єкту.

Детальна інформація про їх структури (ім'я, тип і розмір, опис) приведена у таблицях 4.1 – 4.4.

Таблиця 4.1 включає в себе дані про ризики які мали найбільший вплив на економіку у певні роки.

Таблиця 4.1. Структура таблиці “Топ ризиків за впливом”

Ім'я поля	Тип і розмір поля	Опис поля
Id	int	Первинний ключ
Year	int	Рік
Top1	text	Топ-1 ризик для певного року
Top2	text	Топ-2 ризик для певного року
Top3	text	Топ-3 ризик для певного року
Top4	text	Топ-4 ризик для певного року
Top5	text	Топ-5 ризик для певного року

Таблиця 4.2 включає в себе дані про ризики які мали найбільший шанс проявитися у певні роки.

Таблиця 4.2. Структура таблиці “Топ ризиків за вірогідності прояву”

Ім'я поля	Тип і розмір поля	Опис поля
Id	int	Первинний ключ
Year	int	Рік
Top1	text	Топ-1 ризик для певного року
Top2	text	Топ-2 ризик для певного року

Top3	text	Топ-3 ризик для певного року
Top4	text	Топ-4 ризик для певного року
Top5	text	Топ-5 ризик для певного року

Таблиця 4.3 включає в себе дані про ризики які притаманні певній території і мають найбільший шанс проявитися там.

Таблиця 4.3. Структура таблиці “Топ ризиків за територіями”

Ім'я поля	Тип і розмір поля	Опис поля
Id	int	Первинний ключ
Territory_name	text	Назва території
Top1	text	Топ-1 ризик для певної території
Top2	text	Топ-2 ризик для певної території
Top3	text	Топ-3 ризик для певної території

Таблиця 4.4 включає в себе дані про всі ризики наведені в системі та опис кожного з них.

Таблиця 4.4. Структура таблиці “Опис ризиків”

Ім'я поля	Тип і розмір поля	Опис поля
Id	PK, int	Первинний ключ
Risk_name	text	Назва ризику
Explanation	text	Опис ризику

При проектуванні є дуже важливим детальний опис кожної таблиці та створення їх концептуальної схеми, для подальшого зручного аналізу.

5. РОБОТА КОРИСТУВАЧА З ПРОГРАМНОЮ СИСТЕМОЮ

В цьому розділі приведені системні вимоги для роботи з додатком та сценарії роботи користувача з ним.

5.1 Опис системних вимог

Для створення бази даних було вибрано СКБД MySQL. Це пояснюється тим, що дана СКБД є однією із найпоширеніших і доскональних. Щоб реалізувати інтерфейс користувача, була використана мова програмування Python, та її бібліотека Tkinter. Для інтеграції бази даних використовувалася бібліотека MySQL-connector.

Виходячи з використаних інструментів, для запуску цього комп'ютерного додатку, потрібно мати встановлені: MySQL, Python версії 3.7 або вище, та бібліотеку MySQL-connector.

5.2 Робота користувача з продуктом

На головному, початковому вікні знаходяться кнопки вибору таблиці для дослідження (рисунок 5.1):

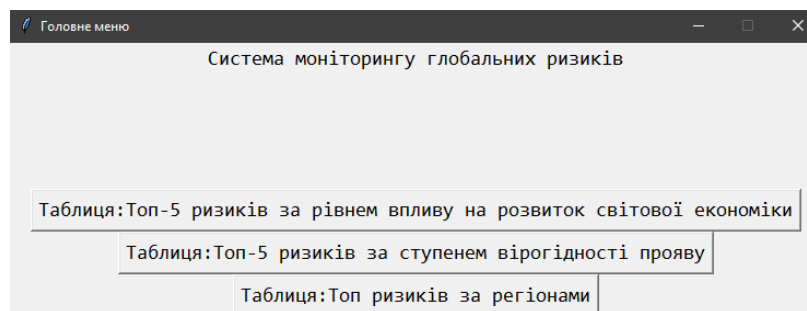


Рисунок 5.1 – Початкове вікно

При натисканні кнопки «Топ-5 ризиків за рівнем впливу на розвиток світової економіки» відкривається вікно з статистичними даними починаючи з 2007 року (рисунок 5.2).

Рік	Перший	Другий	Третій	Четвертий	П'ятий
2007	Колас вартості активів	Скорочення видатків внаслідок	Міждержавні війни з глобальним	Пандемія	Різка зміна цін на нафту
2008	Колас вартості активів	Скорочення видатків внаслідок	Зниження темпів зростання екон	Різка зміна цін на нафту	Пандемія
2009	Колас вартості активів	Скорочення видатків внаслідок	Різка зміна цін на нафту	Хронічні захворювання	Фіскальна криза
2010	Колас вартості активів	Скорочення видатків внаслідок	Різка зміна цін на нафту	Хронічні захворювання	Фіскальна криза
2011	Фіскальна криза	Зміна клімату	Міждержавні війни з глобальним	Колас вартості активів	Шоківі стрибки ціни на енергію
2012	Системні фінансові провали	Криза водних ресурсів	Продовольча криза	Хронічні фіскальні диспропорції	Шоківі стрибки ціни на енергію
2013	Системні фінансові провали	Криза водних ресурсів	Хронічні фіскальні диспропорції	Зброя масового ураження	Провали заходів пом'якшення на
2014	Фіскальна криза	Зміна клімату	Криза водних ресурсів	Безробіття і неповна зайнятість	Руйнування критичної інформації
2015	Криза водних ресурсів	Хронічні захворювання	Зброя масового ураження	Міждержавні війни з глобальним	Провали заходів пом'якшення на
2016	Провали заходів пом'якшення на	Зброя масового ураження	Криза водних ресурсів	Широкомасштабна вимушена мігра	Шоківі стрибки ціни на енергію
2017	Зброя масового ураження	Екстримальні погодні явища	Криза водних ресурсів	Значні природні катастрофи	Провали заходів пом'якшення на
2018	Зброя масового ураження	Екстримальні погодні явища	Значні природні катастрофи	Провали заходів пом'якшення на	Криза водних ресурсів
2019	Зброя масового ураження	Провали заходів пом'якшення на	Екстримальні погодні явища	Криза водних ресурсів	Значні природні катастрофи
2020	Провали заходів пом'якшення на	Зброя масового ураження	Значні втрати біорізноманіття	Екстримальні погодні явища	Криза водних ресурсів

Обрати рік

Рисунок 5.2 – Вікно: «Топ-5 ризиків за рівнем впливу на розвиток світової економіки»

Користувач може вибрати будь який рік для дослідження (рисунок 5.3).

Рік	Перший	Другий	Третій	Четвертий	П'ятий
2007	Колас вартості активів	Скорочення видатків внаслідок	Міждержавні війни з глобальним	Пандемія	Різка зміна цін на нафту
2008	Колас вартості активів	Скорочення видатків внаслідок	Зниження темпів зростання екон	Різка зміна цін на нафту	Пандемія
2009	Колас вартості активів	Скорочення видатків внаслідок	Різка зміна цін на нафту	Хронічні захворювання	Фіскальна криза
2010	Колас вартості активів	Скорочення видатків внаслідок	Різка зміна цін на нафту	Хронічні захворювання	Фіскальна криза
2011	Фіскальна криза	Зміна клімату	Міждержавні війни з глобальним	Колас вартості активів	Шоківі стрибки ціни на енергію
2012	Системні фінансові провали	Криза водних ресурсів	Продовольча криза	Хронічні фіскальні диспропорції	Шоківі стрибки ціни на енергію
2013	Системні фінансові провали	Криза водних ресурсів	Хронічні фіскальні диспропорції	Зброя масового ураження	Провали заходів пом'якшення на
2014	Фіскальна криза	Зміна клімату	Криза водних ресурсів	Безробіття і неповна зайнятість	Руйнування критичної інформації
2015	Криза водних ресурсів	Хронічні захворювання	Зброя масового ураження	Міждержавні війни з глобальним	Провали заходів пом'якшення на
2016	Провали заходів пом'якшення на	Зброя масового ураження	Криза водних ресурсів	Широкомасштабна вимушена мігра	Шоківі стрибки ціни на енергію
2017	Зброя масового ураження	Екстримальні погодні явища	Криза водних ресурсів	Значні природні катастрофи	Провали заходів пом'якшення на
2018	Зброя масового ураження	Екстримальні погодні явища	Значні природні катастрофи	Провали заходів пом'якшення на	Криза водних ресурсів
2019	Зброя масового ураження	Провали заходів пом'якшення на	Екстримальні погодні явища	Криза водних ресурсів	Значні природні катастрофи
2020	Провали заходів пом'якшення на	Зброя масового ураження	Значні втрати біорізноманіття	Екстримальні погодні явища	Криза водних ресурсів

Обрати рік

Рисунок 5.3 – Вікно: «Топ-5 ризиків за рівнем впливу на розвиток світової економіки» з обраним роком

При виборі строки та натисканні кнопки «Обрати рік» відкривається нове вікно з повною інформацією про обраний рік (рисунок 5.4):

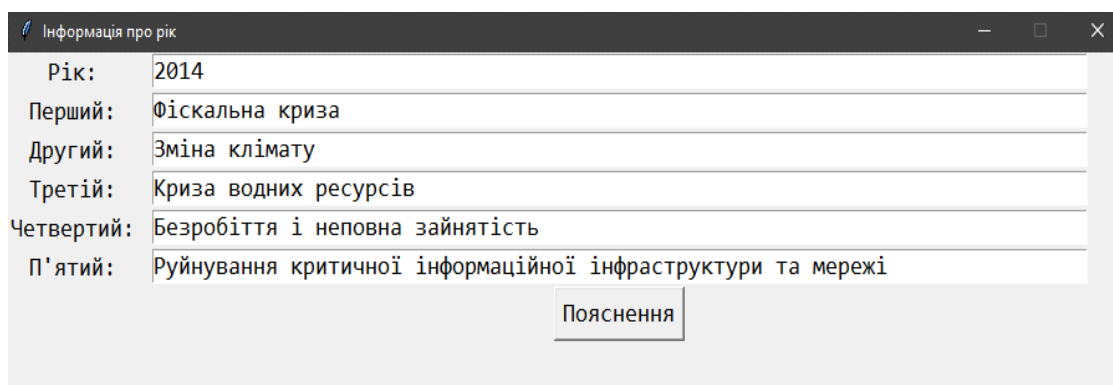


Рисунок 5.4 – Вікно: «Інформація про рік»

Користувач може вибрати будь ризик для дослідження (рисунок 5.5):

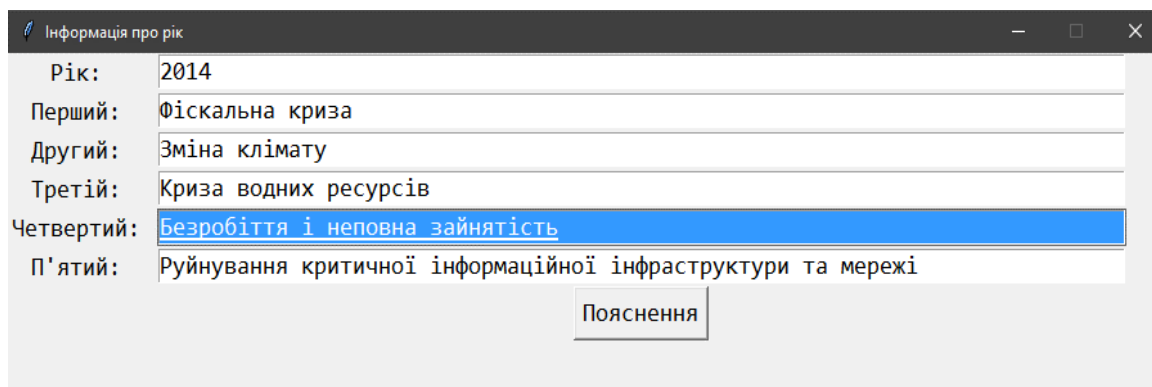


Рисунок 5.5 – Вікно: «Інформація про рік» з обраним ризиком

При виборі ризику та натисканні кнопки «Пояснення» відкривається нове вікно з повною інформацією про обраний ризик (рисунок 5.6):

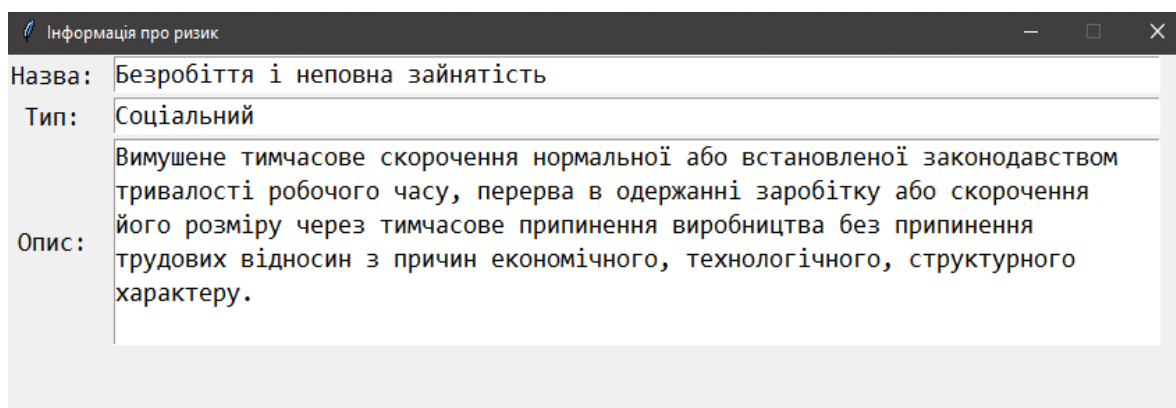


Рисунок 5.6 – Вікно: «Інформація про ризик»

При поверненні на початкове вікно та натисканні кнопки «Топ-5 ризиків за ступенем вірогідності прояву» відкривається вікно з статистичними даними починаючи з 2007 року (рисунок 5.7):

Рік	Перший	Другий	Третій	Четвертий	П'ятий
2007	Руйнування критичної інформації	Хронічні захворювання	Різка зміна цін на нафту	Зниження темпів зростання екон	Колас вартості активів
2008	Колас вартості активів	Зниження темпів зростання екон	Неспроможність національного у	Різка зміна цін на нафту	Хронічні захворювання
2009	Колас вартості активів	Зниження темпів зростання екон	Хронічні захворювання	Неспроможність регіонального ч	Скорочення видатків внаслідок
2010	Колас вартості активів	Зниження темпів зростання екон	Хронічні захворювання	Фіскальна криза	Неспроможність регіонального ч
2011	Значні природні катастрофи	Екстримальні погодні явища	Неспроможність національного у	Значні втрати біорізноманіття	Зміна клімату
2012	Продовольча криза	Хронічні фіскальні диспропорції	Викиди парникових газів	Масштабні кібератаки	Криза водних ресурсів
2013	Продовольча криза	Хронічні фіскальні диспропорції	Викиди парникових газів	Криза водних ресурсів	Старіння населення
2014	Продовольча криза	Екстримальні погодні явища	Безробіття і неповна зайнятіст	Провали заходів пом'якшення на	Масштабні кібератаки
2015	Міждержавні війни з глобальним	Екстримальні погодні явища	Неспроможність національного у	Скорочення видатків внаслідок	Безробіття і неповна зайнятіст
2016	Широкомасштабна вимушена мігра	Екстримальні погодні явища	Провали заходів пом'якшення на	Міждержавні війни з глобальним	Значні природні катастрофи
2017	Екстримальні погодні явища	Широкомасштабна вимушена мігра	Значні природні катастрофи	Масштабні терористичні атаки	Масові випадки шахрайства / кр
2018	Екстримальні погодні явища	Значні природні катастрофи	Масштабні кібератаки	Масові випадки шахрайства / кр	Провали заходів пом'якшення на
2019	Екстримальні погодні явища	Провали заходів пом'якшення на	Значні природні катастрофи	Масові випадки шахрайства / кр	Масштабні кібератаки
2020	Екстримальні погодні явища	Провали заходів пом'якшення на	Значні природні катастрофи	Значні втрати біорізноманіття	Техногенні збитки навколишньом

Рисунок 5.7 – Вікно: «Топ-5 ризиків за ступенем вірогідності прояву»

Користувач може вибрати будь який рік для дослідження (рисунок 5.8).

Рік	Перший	Другий	Третій	Четвертий	П'ятий
2007	Руйнування критичної інформації	Хронічні захворювання	Різка зміна цін на нафту	Зниження темпів зростання екон	Колас вартості активів
2008	Колас вартості активів	Зниження темпів зростання екон	Неспроможність національного у	Різка зміна цін на нафту	Хронічні захворювання
2009	Колас вартості активів	Зниження темпів зростання екон	Хронічні захворювання	Неспроможність регіонального ч	Скорочення видатків внаслідок
2010	Колас вартості активів	Зниження темпів зростання екон	Хронічні захворювання	Фіскальна криза	Неспроможність регіонального ч
2011	Значні природні катастрофи	Екстримальні погодні явища	Неспроможність національного у	Значні втрати біорізноманіття	Зміна клімату
2012	Продовольча криза	Хронічні фіскальні диспропорції	Викиди парникових газів	Масштабні кібератаки	Криза водних ресурсів
2013	Продовольча криза	Хронічні фіскальні диспропорції	Викиди парникових газів	Криза водних ресурсів	Старіння населення
2014	Продовольча криза	Екстримальні погодні явища	Безробіття і неповна зайнятіст	Провали заходів пом'якшення на	Масштабні кібератаки
2015	Міждержавні війни з глобальним	Екстримальні погодні явища	Неспроможність національного у	Скорочення видатків внаслідок	Безробіття і неповна зайнятіст
2016	Широкомасштабна вимушена мігра	Екстримальні погодні явища	Провали заходів пом'якшення на	Міждержавні війни з глобальним	Значні природні катастрофи
2017	Екстримальні погодні явища	Широкомасштабна вимушена мігра	Значні природні катастрофи	Масштабні терористичні атаки	Масові випадки шахрайства / кр
2018	Екстримальні погодні явища	Значні природні катастрофи	Масштабні кібератаки	Масові випадки шахрайства / кр	Провали заходів пом'якшення на
2019	Екстримальні погодні явища	Провали заходів пом'якшення на	Значні природні катастрофи	Масові випадки шахрайства / кр	Масштабні кібератаки
2020	Екстримальні погодні явища	Провали заходів пом'якшення на	Значні природні катастрофи	Значні втрати біорізноманіття	Техногенні збитки навколишньом

Рисунок 5.8 – Вікно: «Топ-5 ризиків за ступенем вірогідності прояву» з обраним роком

При виборі строки та натисканні кнопки «Обрати рік» відкривається нове вікно з повною інформацією про обраний рік (рисунок 5.9).

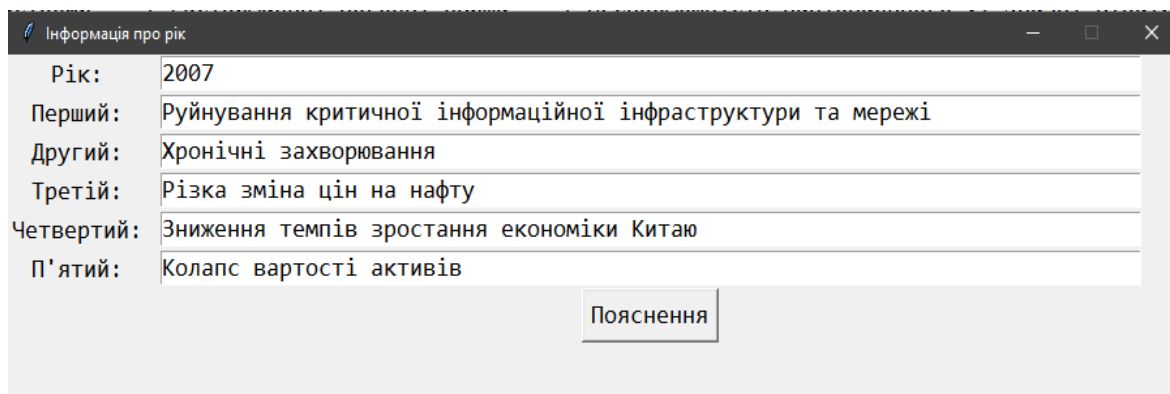


Рисунок 5.9 – Вікно: «Інформація про рік»

Користувач може вибрати будь ризик для дослідження (рисунок 5.10).

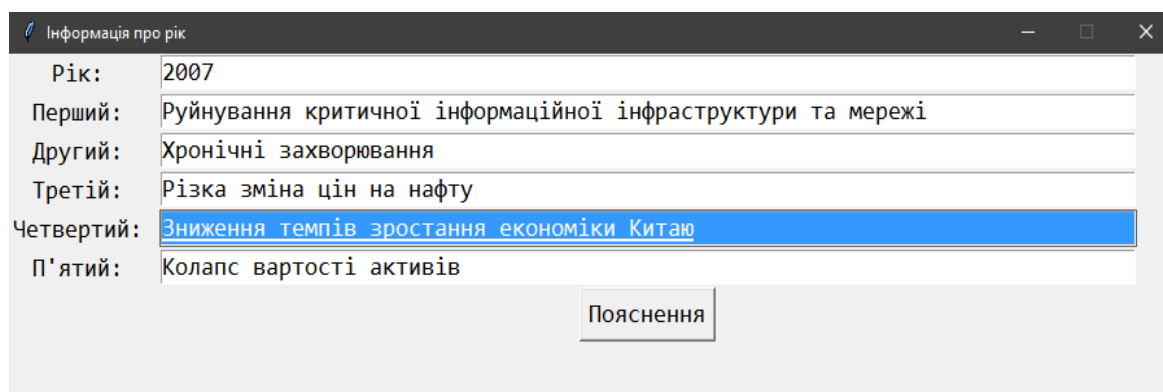


Рисунок 5.10 – Вікно: «Інформація про рік» з обраним ризиком

При виборі ризику та натисканні кнопки «Пояснення» відкривається нове вікно з повною інформацією про обраний ризик (рисунок 5.11).

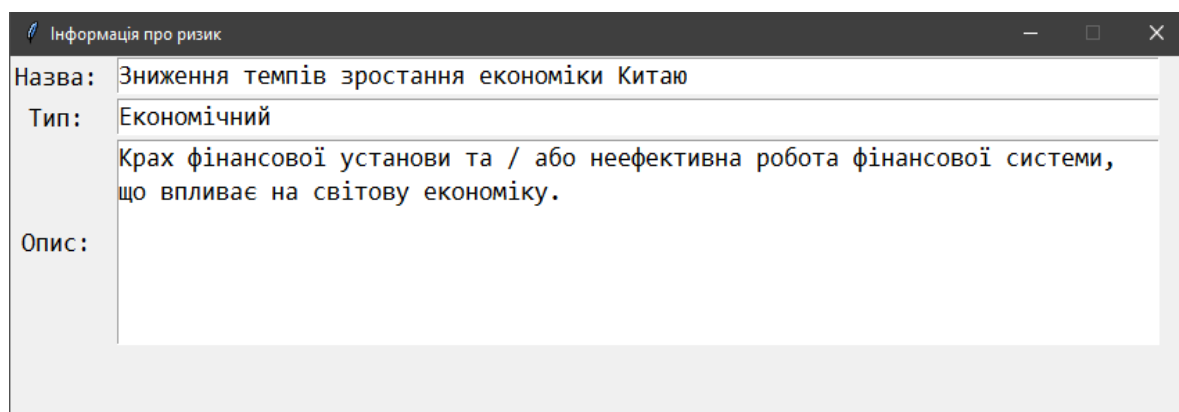


Рисунок 5.11 – Вікно: «Інформація про ризик»

При поверненні на початкове вікно та натисканні кнопки «Топ ризиків за регіонами» відкривається вікно з статистичними даними про ризики які притаманні певному регіону (рисунок 5.12).

Топ ризиків за регіонами			
Регіон	Перший	Другий	Третій
Європа	Безробіття і неповна зайнятість	Широкомасштабна вимушена міграція	Значна соціальна нестабільність
Середньозахідна та Північна Африка	Криза водних ресурсів	Значна соціальна нестабільність	Міждержавні війни з глобальними наслідками
Південна Африка	Безробіття і неповна зайнятість	Продовольча криза	Хронічні захворювання
Північна Америка	Масштабні кібератаки	Нездатність міського планування	Значна соціальна нестабільність
Латинська Америка	Значна соціальна нестабільність	Неспроможність національного управління	Незаконна торгівля
Центральна Азія	Різка зміна цін на нафту	Масштабні терористичні атаки	Міждержавні війни з глобальними наслідками
Азійсько-Тихоокеанський регіон	Міждержавні війни з глобальними наслідками	Значні природні катастрофи	Викиди парникових газів
Південна Азія	Криза водних ресурсів	Масштабні терористичні атаки	Продовольча криза

Рисунок 5.12 – Вікно: «Топ ризиків за регіонами»

ВИСНОВКИ

В ході виконання даної роботи було розроблено багатопараметричну систему моніторингу рівня глобальних ризиків.

Програму було написано на мові програмування Python з використанням графічного інтерфейсу бібліотеки Tkinter.

Програму можна використовувати для дослідження рівня глобальних ризиків за роками, на основі статистичних даних.

В ході роботи були використані такі утиліти для створення даного програмного забезпечення (середовища розробки Jupyter Notebook, MySQL Workbench).

Для роботи з даним програмним забезпеченням необхідний лише комп'ютер середньої потужності.

Користувачу буде надана можливість опрацьовувати дані різних показників за останні роки та регіони, прогнозувати варіанти подальшого розвитку ризиків та аналізувати які превентивні методи слід використати.

Програма дає можливість доповнювати базу даних статистичними показниками за новий період.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. WeForum. Global risk report 2020 [Електронний ресурс] / WeForum – Режим доступу до ресурсу : <https://www.weforum.org/reports/the-global-risks-report-2020> (дата звернення 15.05.2021).
2. WeForum. Global risk report 2019 [Електронний ресурс] / WeForum – Режим доступу до ресурсу : <https://www.weforum.org/reports/the-global-risks-report-2019> (дата звернення 15.05.2021).
3. WeForum. Global risk report 2016 [Електронний ресурс] / WeForum – Режим доступу до ресурсу : <https://www.weforum.org/reports/the-global-risks-report-2016> (дата звернення 15.05.2021).
4. Мэттиз Э. Изучаем Python. Программирование игр, визуализация данных, веб-приложения / Эрик Мэттиз. – Санкт-Петербург: Питер, 2020. – 600 с.
5. Бэрри П. Изучаем программирование на Python / Пол Бэрри. – Москва: Эксмо, 2017. – 624 с.
6. Любанович Б. Простой Python. Современный стиль программирования / Билл Любанович. – Санкт-Петербург: Питер, 2020. – 592 с.
7. Рамальо Л. Python. К вершинам мастерства / Лучано Рамальо. – Москва: ДМК Пресс, 2016. – 768 с.
8. Хеллман Д. Стандартная библиотека Python 3 справочник с примерами / Даг Хеллман. – Москва: Вильямс, 2017. – 320 с.
9. Персиваль Г. Python. Разработка на основе тестирования / Гарри Персиваль. – Москва: ДМК Пресс, 2018. – 624 с.
10. Лутц М. Программирование на Python / Марк Лутц. – Москва: Символ-плюс, 2011. – 990 с.
11. Бизли Д. Python. Подробный справочник / Дэвид Бизли. – Москва: Символ-плюс, 2010. – 850 с.
12. Создание графического интерфейса Tkinter [Електронний ресурс] – Режим доступу до ресурсу : <https://metanit.com/python/tutorial/9.1.php>. (дата звернення 23.04.2021).

13. Дюбуа П. MySQL. Сборник рецептов. Руководство / Поль Дюбуа. – Москва: Символ-плюс, 2017. – 1056 с.
14. Шварц Б. MySQL по максимуму / Б. Шварц, П. Зайцев, В. Ткаченко. – Санкт-Петербург: Питер, 2018. – 864 с.
15. Пособие по MySQL на Python [Электронный ресурс] – Режим доступа до ресурсу: <https://www.internet-technologies.ru/articles/posobie-po-mysql-na-python.html> (дата звернення 23.04.2021).
16. Гарсиа Г. Системы баз данных / Г. Гарсиа, Д. Ульман, Д. Уидом. – Москва: Вильямс, 2016. – 1088 с.
17. Уэнц К. PHP и MySQL. Карманный справочник / Кристиан Уэнц. – Москва: Вильямс, 2013. – 256 с.
18. Тахагхогхи С. Руководство по MySQL / Сейед Тахагхогхи. – Москва: Русская Редакция, 2013. – 544 с.
19. Алхабри Д. MySQL. Справочник по языку / Джозеф Алхабри. – Москва: Вильямс, 2005. – 432 с.
20. MySQL. Оптимизация производительности / [Ш. Берон, П. Зайцев, В. Ткаченко та ін.]. – Москва: Символ-плюс, 2010. – 832 с.
21. Лесик М.О. Система моніторингу рівня глобальних ризиків / М.О. Лесик, Н.В. Караєва / Матеріали ХІХ Міжнар. наук.-практ. конф. молод. вчених і студ. «Сучасні проблеми наукового забезпечення енергетики», м. Київ, 20–23 квіт. 2021 р. – Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2021. – Т. 2. – С. 280-281.

ДОДАТОК А

Система моніторингу рівня глобальних ризиків

Специфікація

УКР.НТУУ”КПІ”_ТЕФ_АПЕПС_ТМ 72195_21Б

Аркушів 2

Київ — 2021

Позначення	Найменування	Примітки
Документація		
УКР.НТУУ"КП" _ТЕФ_АПЕПС_ 72195_21Б	Записка.docx	Текстова частина дипломної роботи
Компоненти		
УКР.НТУУ"КП" _ТЕФ_АПЕПС_ 72195_21Б 12-1	main.ipynb	Основний файл системи, що містить інтерфейс та проводить операції з БД

ДОДАТОК Б

Система моніторингу рівня глобальних ризиків

Текст програми

УКР.НТУУ"КПІ" _ТЕФ_АПЕПС_ТМ 72195_21Б

Аркушів 16

Київ — 2021

```
import mysql.connector
from tkinter import *
from tkinter.font import Font

mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    passwd="max11245",
    database="globalriskreport"
)

print(mydb)

mycursor = mydb.cursor()

mycursor.execute("SHOW DATABASES")

for x in mycursor:
    print(x)

mycursor = mydb.cursor()

mycursor.execute(" SELECT * FROM
globalriskreport.top5_in_terms_of_impact;")

myresult = mycursor.fetchall()
```

```

def error_message():
    def destroy(event):
        err_window.destroy()
    err_window=Toplevel()
    err_window.resizable(False, False)
    err_window.geometry("300x100+800+500")
    err_window.title("Помилка")
    err_label=Label(err_window,text="Помилка. Не вибрано поле.",font=font_2)
    btn_err= Button(err_window,text="ОК",font=font_2)
    btn_err.bind('<Button-1>', destroy)
    err_label.pack()
    btn_err.pack()

def table_impact(event):
    def year_info(event):
        def expl(event):
            risk_name=""
            if(top1_listbox.curselection() !=()):
                risk_name=myresult1[ind_1[0]-1][1]
            if(top2_listbox.curselection() !=()):
                risk_name=myresult1[ind_1[0]-1][2]
            if(top3_listbox.curselection() !=()):
                risk_name=myresult1[ind_1[0]-1][3]
            if(top4_listbox.curselection() !=()):
                risk_name=myresult1[ind_1[0]-1][4]

```

```

if(top5_listbox.curselection() !=()):
    risk_name=myresult1[ind_1[0]-1][5]
if(risk_name == ""):
    error_message()
else:
    expl_window=Toplevel()
    expl_window.geometry("850x250+500+400")
    expl_window.resizable(False, False)
    expl_window.title("Інформація про ризик")

    name_listbox=Listbox(expl_window, height=1,width=75,font=font_2)

    type_listbox=Listbox(expl_window, height=1,width=75,font=font_2)
    expl_listbox=Listbox(expl_window, height=6,width=75,font=font_2)

    name_label=Label(expl_window,text="Назва: ",font=font_2)
    type_label=Label(expl_window,text="Тип: ",font=font_2)
    expl_label=Label(expl_window,text="Опис: ",font=font_2)
    #window_label=Label(expl_window,text="Назва: ",font=font_2)

    mycursor.execute('select * from globalriskreport.risk_explanation where
risk_name= "{0}";'.format(risk_name))
    expl_arr=mycursor.fetchall()
    #for y in expl_arr:
        #print(y)
    name_listbox.insert(END,risk_name)
    type_listbox.insert(END,'{0:50}'.format (str(expl_arr[0][3])))

```

```

s=str(expl_arr[0][2])
ln=75
#print(len(s))
while len(s) > ln:
    space_id = s.rindex(' ', 0, ln)
    expl_listbox.insert(END,s[0: space_id])
    s = s[space_id + 1: len(s)]
expl_listbox.insert(END,s)
#expl_listbox.insert(END,' {0:5}'.format (str(expl_arr[0][2])))

name_label.grid(row=0,column=0)
name_listbox.grid(row=0,column=1)

type_label.grid(row=1,column=0)
type_listbox.grid(row=1,column=1)

expl_label.grid(row=2,column=0)
expl_listbox.grid(row=2,column=1)

#name_listbox.pack()
#type_listbox.pack()
#expl_listbox.pack()
ind_1=list(myresult_listbox.curselection())
if(myresult_listbox.curselection()==()):
    error_message()
else:

```

```
year_window=Toplevel()
year_window.geometry("900x250+500+400")
year_window.resizable(False, False)
year_window.title("Інформація про рік")

print(myresult_listbox.curselection())

label=Label(year_window,text="Рік: ",font=font_2)
year_listbox=Listbox(year_window,height=1,width=75,font=font_2)
year_listbox.insert(END,myresult1[ind_1[0]-1][0])

label1=Label(year_window,text="Перший: ",font=font_2)
top1_listbox=Listbox(year_window,height=1,width=75,font=font_2)
top1_listbox.insert(END,myresult1[ind_1[0]-1][1])

label2=Label(year_window,text="Другий: ",font=font_2)
top2_listbox=Listbox(year_window,height=1,width=75,font=font_2)
top2_listbox.insert(END,myresult1[ind_1[0]-1][2])

label3=Label(year_window,text="Третій: ",font=font_2)
top3_listbox=Listbox(year_window,height=1,width=75,font=font_2)
top3_listbox.insert(END,myresult1[ind_1[0]-1][3])

label4=Label(year_window,text="Четвертий: ",font=font_2)
top4_listbox=Listbox(year_window,height=1,width=75,font=font_2)
top4_listbox.insert(END,myresult1[ind_1[0]-1][4])

label5=Label(year_window,text="П'ятий: ",font=font_2)
```

```
top5_listbox=Listbox(year_window,height=1,width=75,font=font_2)
```

```
top5_listbox.insert(END,myresult1[ind_1[0]-1][5])
```

```
label.grid(row=0,column=0)
```

```
year_listbox.grid(row=0,column=2)
```

```
label1.grid(row=2,column=0)
```

```
top1_listbox.grid(row=2,column=2)
```

```
label2.grid(row=4,column=0)
```

```
top2_listbox.grid(row=4,column=2)
```

```
label3.grid(row=6,column=0)
```

```
top3_listbox.grid(row=6,column=2)
```

```
label4.grid(row=8,column=0)
```

```
top4_listbox.grid(row=8,column=2)
```

```
label5.grid(row=10,column=0)
```

```
top5_listbox.grid(row=10,column=2)
```

```
btn_exp= Button(year_window,text="Пояснения",font=font_2)
```

```
btn_exp.bind('<Button-1>',expl)
```

```
btn_exp.grid(row=12,column=2)
```

```

window = Toplevel()
window.geometry("1700x600")
window.resizable(False, False)
window.title("Топ-5 ризиків за рівнем впливу на розвиток світової економіки")

mycursor.execute("select year,top1,top2,top3,top4,top5 from
globalriskreport.top5_in_terms_of_impact;")

myresult1 = mycursor.fetchall()

myresult_listbox = Listbox(window,height = 15, width=168,font=font_1)
label_=Label(window,text="Топ-5 ризиків за рівнем впливу на розвиток світової економіки", font=font_1)
label_.pack()

shapka='| Рік | Перший | Другий | Третій
| Четвертий | П'ятий |'
myresult_listbox.insert(END,shapka)
for i in myresult:
    myresult_listbox.insert(END,' {0:5}| {1:30}| {2:30}| {3:30}| {4:30}|
{5:30}|'.format(str(i[1]),
str(i[2][:30]),str(i[3][:30]),str(i[4][:30]),str(i[5][:30]),str(i[6][:30])))

myresult_listbox.pack()
year=StringVar()
btn= Button(window,text="Обрати рік",font=font_2)
btn.pack()
btn.bind('<Button-1>',year_info)

def table_likelihood(event):
    def year_info(event):

```

```
def expl(event):
    risk_name=""
    if(top1_listbox.curselection() !=()):
        risk_name=myresult1[ind_1[0]-1][1]
    if(top2_listbox.curselection() !=()):
        risk_name=myresult1[ind_1[0]-1][2]
    if(top3_listbox.curselection() !=()):
        risk_name=myresult1[ind_1[0]-1][3]
    if(top4_listbox.curselection() !=()):
        risk_name=myresult1[ind_1[0]-1][4]
    if(top5_listbox.curselection() !=()):
        risk_name=myresult1[ind_1[0]-1][5]
    if(risk_name == ""):
        error_message()
    else:
        expl_window=Toplevel()
        expl_window.geometry("850x250+500+400")
        expl_window.resizable(False, False)
        expl_window.title("Інформація про ризик")

        name_listbox=Listbox(expl_window, height=1,width=75,font=font_2)

        type_listbox=Listbox(expl_window, height=1,width=75,font=font_2)
        expl_listbox=Listbox(expl_window, height=6,width=75,font=font_2)

        name_label=Label(expl_window,text="Назва: ",font=font_2)
        type_label=Label(expl_window,text="Тип: ",font=font_2)
        expl_label=Label(expl_window,text="Опис: ",font=font_2)
```

```
#window_label=Label(expl_window,text="Назва: ",font=font_2)
```

```
mycursor.execute('select * from globalriskreport.risk_explanation where
risk_name= "{0}";'.format(risk_name))
```

```
expl_arr=mycursor.fetchall()
```

```
#for y in expl_arr:
```

```
    #print(y)
```

```
name_listbox.insert(END,risk_name)
```

```
type_listbox.insert(END,'{0:50}'.format (str(expl_arr[0][3])))
```

```
s=str(expl_arr[0][2])
```

```
ln=75
```

```
#print(len(s))
```

```
while len(s) > ln:
```

```
    space_id = s.rindex(' ', 0, ln)
```

```
    expl_listbox.insert(END,s[0: space_id])
```

```
    s = s[space_id + 1: len(s)]
```

```
expl_listbox.insert(END,s)
```

```
#expl_listbox.insert(END,' {0:5}'.format (str(expl_arr[0][2])))
```

```
name_label.grid(row=0,column=0)
```

```
name_listbox.grid(row=0,column=1)
```

```
type_label.grid(row=1,column=0)
```

```
type_listbox.grid(row=1,column=1)
```

```
expl_label.grid(row=2,column=0)
```

```
expl_listbox.grid(row=2,column=1)

ind_1=list(myresult_listbox.curselection())
if(myresult_listbox.curselection()==()):
    error_message()
else:

    year_window=Toplevel()
    year_window.geometry("900x250+500+400")
    year_window.resizable(False, False)
    year_window.title("Інформація про рік")

    print(myresult_listbox.curselection())

    label=Label(year_window,text="Рік: ",font=font_2)
    year_listbox=Listbox(year_window,height=1,width=75,font=font_2)
    year_listbox.insert(END,myresult1[ind_1[0]-1][0])

    label1=Label(year_window,text="Перший: ",font=font_2)
    top1_listbox=Listbox(year_window,height=1,width=75,font=font_2)
    top1_listbox.insert(END,myresult1[ind_1[0]-1][1])

    label2=Label(year_window,text="Другий: ",font=font_2)
    top2_listbox=Listbox(year_window,height=1,width=75,font=font_2)
    top2_listbox.insert(END,myresult1[ind_1[0]-1][2])

    label3=Label(year_window,text="Третій: ",font=font_2)
    top3_listbox=Listbox(year_window,height=1,width=75,font=font_2)
```

```
top3_listbox.insert(END,myresult1[ind_1[0]-1][3])
```

```
label4=Label(year_window,text="Четвертый: ",font=font_2)
```

```
top4_listbox=Listbox(year_window,height=1,width=75,font=font_2)
```

```
top4_listbox.insert(END,myresult1[ind_1[0]-1][4])
```

```
label5=Label(year_window,text="П\`ятый: ",font=font_2)
```

```
top5_listbox=Listbox(year_window,height=1,width=75,font=font_2)
```

```
top5_listbox.insert(END,myresult1[ind_1[0]-1][5])
```

```
label.grid(row=0,column=0)
```

```
year_listbox.grid(row=0,column=2)
```

```
label1.grid(row=2,column=0)
```

```
top1_listbox.grid(row=2,column=2)
```

```
label2.grid(row=4,column=0)
```

```
top2_listbox.grid(row=4,column=2)
```

```
label3.grid(row=6,column=0)
```

```
top3_listbox.grid(row=6,column=2)
```

```
label4.grid(row=8,column=0)
```

```
top4_listbox.grid(row=8,column=2)
```

```
label5.grid(row=10,column=0)
```

```
top5_listbox.grid(row=10,column=2)
```

```

btn_exp= Button(year_window,text="Пояснення",font=font_2)
btn_exp.bind('<Button-1>',expl)
btn_exp.grid(row=12,column=2)

```

```

window = Toplevel()
window.geometry("1700x600")
window.resizable(False, False)
window.title("Топ-5 ризиків за ступенем вірогідності прояву")
mycursor.execute("SELECT * FROM
globalriskreport.top5_in_terms_of_likelihood;")

```

```

myresult = mycursor.fetchall()
mycursor.execute("select year,top1,top2,top3,top4,top5 from
top5_in_terms_of_likelihood;")
myresult1 = mycursor.fetchall()

```

```

myresult_listbox = Listbox(window,height = 15, width=168,font=font_1)
label_=Label(window,text="Топ-5 ризиків за ступенем вірогідності прояву",
font=font_1)
label_.pack()
shapka='| Рік | Перший | Другий | Третій
| Четвертий | П'ятий |'
myresult_listbox.insert(END,shapka)
for i in myresult:

```

```

myresult_listbox.insert(END,' {0:5}| {1:30}| {2:30}| {3:30}| {4:30}|
{5:30}|'.format(str(i[1]),
str(i[2][:30]),str(i[3][:30]),str(i[4][:30]),str(i[5][:30]),str(i[6][:30])))

```

```

myresult_listbox.pack()
year=StringVar()
btn= Button(window,text="Обрати рік",font=font_2)
btn.pack()
btn.bind('<Button-1>',year_info)

```

```

def table_region(event):

```

```

    window = Toplevel()
    window.geometry("1700x300")
    window.title("Топ ризиків за регіонами")
    window.resizable(False, False)

```

```

    mycursor.execute("SELECT * FROM globalriskreport.risks_by_territory;")

```

```

    myresult1 = mycursor.fetchall()

```

```

    myresult_listbox = Listbox(window,height = 9, width=168,font=font_1)

```

```

    label_=Label(window,text="Топ ризиків за регіонами", font=font_1)

```

```

    label_.pack()

```

```

    shapka='| Регіон                | Перший                | Другий
| Третій                |'

```

```

    myresult_listbox.insert(END,shapka)

```

```

    for i in myresult1:

```

```

        myresult_listbox.insert(END,' {0:34}| {1:42}| {2:39}|
{3:42}|'.format(str(i[1]), str(i[2]),str(i[3]),str(i[4])))

```

```
myresult_listbox.pack()

root = Tk()
login=0
password=0
root.title("Головне меню")
root.geometry("745x250+500+500")
font_1= Font(family="Consolas" ,underline=True, size=14)
font_2= Font(family="Consolas" ,underline=False, size=14)

top_frame=Frame(root)
top_frame.pack()

bot_frame=Frame(root)
bot_frame.pack(side=BOTTOM)

label1=Label(top_frame,text="Система моніторингу глобальних ризиків",
font=font_2)

btn = Button(bot_frame,text="Таблиця:Топ-5 ризиків за рівнем впливу на
розвиток світової економіки" , font=font_2 )

btn1= Button(bot_frame,text = "Таблиця:Топ-5 ризиків за ступенем
вірогідності прояву", font=font_2)

btn2= Button(bot_frame,text = "Таблиця:Топ ризиків за регіонами",
font=font_2)

label1.pack()

btn.pack()

btn1.pack()

btn2.pack()

btn.bind('<Button-1>', table_impact)
```

```
btn1.bind('<Button-1>',table_likelihoood)
btn2.bind('<Button-1>', table_region)
root.resizable(False, False)
root.mainloop()
```

ДОДАТОК В

Система моніторингу рівня глобальних ризиків

Опис програми

УКР.НТУУ"КПІ" _ТЕФ_АПЕПС_ТМ 72195_21Б

Аркушів 10

Київ — 2021

АНОТАЦІЯ

Додаток містить опис застосунку системи моніторингу рівня глобальних ризиків, що виконує деякі завдання, визначенні в розділі 1, а саме:

- аналіз динаміки зміни ризиків за роками;
- розподілення найімовірніших ризиків за регіонами;
- візуалізація отриманих результатів у табличному вигляді для подальшого аналізу.
- можливість доповнювати базу даних (БД).

Застосунок розроблений мовою програмування Python з використанням Tkinter у редакторі вихідного коду Jupyter Notebook.

ЗМІСТ

1. ЗАГАЛЬНІ ВІДОМОСТІ	71
2. ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ.....	72
3. ОПИС ЛОГІЧНОЇ СТРУКТУРИ	73
4. ВИКОРИСТОВУВАНІ ТЕХНІЧНІ ЗАСОБИ.....	74
5. ВИКЛИК І ЗАВАНТАЖЕННЯ.....	75
6. ВХІДНІ ДАНІ	76
7. ВИХІДНІ ДАНІ	77

ЗАГАЛЬНІ ВІДОМОСТІ

У додатку міститься опис основних компонентів застосунку системи моніторингу рівня глобальних ризиків, що виконує деякі завдання, визначені в розділі 1. Додаток Б містить програмний код цих компонентів.

Для роботи застосунку потрібно мати комп'ютер з операційною системою Windows не нище 10 версії.

Застосунок розроблений мовою програмування Python з використанням Tkinter у редакторі вихідного коду Jupyter Notebook.

ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ

Розроблений застосунок надає користувачу наступний функціонал:

- аналіз динаміки зміни ризиків за роками;
- розподілення найімовірніших ризиків за регіонами;
- візуалізація отриманих результатів у табличному вигляді для подальшого аналізу.
- можливість доповнювати базу даних (БД).

ОПИС ЛОГІЧНОЇ СТРУКТУРИ

Згідно з архітектурою, застосунок складається з двох частин: бази даних та інтерфейсу.

База даних містить наступні таблиці:

- Таблиця ризиків за впливом;
- Таблиця ризиків за вірогідності прояву;
- Таблиця ризиків за регіоном;
- Таблиця ризиків та їх опис;

Модуль інтерфейсу складається з наступних частин модуль виводу, заміни, та форматування.

Модуль виводу відповідає за відображення таблиць у системі.

Модуль заміни допомагає користувачеві змінювати дані в БД.

Модуль форматування допомагає користувачеві видаляти, замінити та актуалізувати дані БД.

ВИКОРИСТОВУВАНІ ТЕХНІЧНІ ЗАСОБИ

Для роботи застосунку потрібно мати комп'ютер з операційною системою Windows не нище 10 версії.

Для роботи застосунку потрібно мати на комп'ютері Python версії 3.9 або вище, встановлені бібліотеки Tkinter та MySQLconnector, також потрібно мати СКБД Mysql.

ВИКЛИК І ЗАВАНТАЖЕННЯ

Для встановлення застосунку на комп'ютер потрібно просто перенести вже скомпільований додаток та встановити базу даних через MySQL Workbench. Та потім просто запустити .bat файл.

ВХІДНІ ДАНІ

Вхідними даними системи є звіт Всесвітнього економічного форуму Global risk report. Що в свою чергу є щорічним звітом яким WEF виконує поставлену собою ж задачу по повідомленню та розповсюдженню інформації про ризики в маси.

ВИХІДНІ ДАНІ

Вихідні дані системи це довідникові таблиці:

- Таблиця ризиків за ступенем впливу;
- Таблиця ризиків за вірогідністю прояву;
- Таблиця ризиків за регіонами;
- Таблиця ризиків та їх опис;

ДОДАТОК Г

Система моніторингу рівня глобальних ризиків

Апробація

УКР.НТУУ”КПІ”_ТЕФ_АПЕПС_ТМ 72195_21Б

Аркушів 5

Київ — 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ГОРЯ СІКОРСЬКОГО»

СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ

Матеріали ХІХ Міжнародної
науково-практичної конференції
молодих вчених і студентів
м. Київ, 20–23 квітня 2021 року

ТОМ 2



Київ- 2021

<i>Керівник - доц., к.т.н. Левченко Л.О.</i>	
Аналіз веб-платформ дистанційного навчання.	262
<i>КОЛОДА В.Є., магістрант гр. ТВ-301мп</i>	
<i>Керівник - доц., к.т.н. Ходаковський О.В.</i>	
Інструментальні засоби виявлення дорожньо-транспортних пригод.	264
<i>ГРИКУН П.І., магістрант гр. ТВ-301мп</i>	
<i>Керівник - доц., к.е.н. Гусева І.І.</i>	
Характеристика методів нормування показників у контексті розробки системи моніторингу індикаторів сталого розвитку.	266
<i>АГАФОНОВА С.В., магістрант гр. ТВ-301мп</i>	
<i>Керівник - доц., к.е.н. Караєва Н.В.</i>	
Fuzzy logic methods for territory's sustainable development risk assessment .	268
<i>TULUK A.S., master гр. TP-02мп</i>	
<i>Scientific chief - assoc.prof., cand.econ.sc. Karaieva N.V.</i>	
Research of features of influence of electromagnetic fields of ultrahigh and higher frequencies on workers.	270
<i>KOZACHUK A.D., master гр. ТВ-01мп</i>	
<i>Scientific chief - assoc.prof., cand.tech.sc. Khodakovskiy O.V.</i>	
Методика оцінки спалених пожежами територій за космічними знімками.	272
<i>ЧОРНИЙ О.О., студент гр. ТМ-72</i>	
<i>Керівник - ст.викл. Бандурка О.І.</i>	
Інструментальні засоби розпізнавання поведінки за кермом.	274
<i>САРАФАННИКОВ О.В., студент гр. ТМ-71</i>	
<i>Керівник - доц., к.т.н. Гусева І.І.</i>	
Інформаційне забезпечення аналізу часових змін лісових масивів.	276
<i>ОНИЩЕНКО Р.С., студент гр. ТМ-72</i>	
<i>Керівник - ст.викл. Бандурка О.І.</i>	
Підсистема обліку та аналізу стану кадрового забезпечення.	278
<i>ЛУТАЙ Є.Л., студент гр. ПІ-71</i>	
<i>Керівник - асист., к.е.н. Гусева І.І.</i>	
Система моніторингу рівня глобальних ризиків.	280
<i>ЛЕСИК М.О., студент гр. ТМ-72</i>	
<i>Керівник - доц., к.е.н. Караєва Н.В.</i>	
Пошукова GIS система екологічно небезпечних промислових об'єктів.	282
<i>ЛАСКАВИЙ О.О., студент гр. ТМ-71</i>	
<i>Керівник - ст.викл. Бандурка О.І.</i>	
Система кластеризації території України за рівнем екологічної безпеки.	284
<i>ДІБРОВА А.В., студент гр. ТМ-71</i>	
<i>Керівник - ст.викл. Бандурка О.І.</i>	
Інформаційна система оцінки ризику впливу від повенів.	286
<i>БАРАБАШ В.І., студент гр. ТМ-72</i>	
<i>Керівник - ст.викл. Бандурка О.І.</i>	
Recognition tools and data processing of two-dimensional image.	288
<i>СКУВА N.V., master гр. ПІ-72</i>	
<i>Scientific chief - assoc.prof., cand.econ.sc. Husyeva I.I.</i>	
Застосування ІТ-технологій в електроенергетиці.	290
<i>КРИВДА Д.О., студент гр. ТВ-91</i>	
<i>Керівник - доц., к.е.н. Кривда О.В.</i>	

УДК 004.094:368

Студент 4 курсу, гр. ТМ-72 Лесик М.О.
Доц., к.е.н. Караєва Н.В.

СИСТЕМА МОНІТОРИНГУ РІВНЯ ГЛОБАЛЬНИХ РИЗИКІВ

Одним з найбільш тривожних і парадоксальних наслідків глобалізації є її вплив на ризики. Експертами міжнародної організації Всесвітнього економічного форуму (ВЕФ) поняття «глобальний ризик» визначається як невизначена подія чи стан, які у разі їх виникнення можуть спричинити значний негативний вплив для кількох країн або галузей промисловості протягом наступних 10 років [1]. У табл. 1 зведено Топ-5 глобальних ризиків за ступенем вірогідності прояву та з точки зору впливу події, які визначено ВЕФ за період 2018-2020 рр.

Таблиця 1 – Топ-5 глобальних ризиків за різними критеріями

Рік	Місце ризику				
	1	2	3	4	5
<i>за ступенем вірогідності прояву</i>					
2018	екстремальні погодні явища ²	значні природні катастрофи ²	кібер-атаки ¹	масові випадки крадіжки даних/шахрайства ⁵	провали заходів пом'якшення наслідків зміни клімату ²
2019	екстремальні погодні явища ²	провали заходів пом'якшення наслідків зміни клімату ²	значні природні катастрофи ²	масові випадки крадіжки даних/шахрайства ⁵	кібер-атаки ⁵
2020	екстремальні погодні явища ²	провали заходів пом'якшення наслідків зміни клімату ²	значні природні катастрофи ²	втрата біорізноманіття ²	антропогенно природні катастрофи ²
<i>з точки зору впливу події, якщо вона станеться</i>					
2018	зброя масового ураження ³	екстремальні погодні явища ²	природні катастрофи ²	провали заходів пом'якшення наслідків зміни клімату ²	водна криза (водозабезпечення) ⁴
2019	зброя масового ураження ³	провали заходів пом'якшення наслідків зміни клімату ²	екстремальні погодні явища ²	водна криза (водозабезпечення) ⁴	природні катастрофи ²
2020	провали заходів пом'якшення наслідків зміни клімату ²	зброя масового ураження ³	втрата біорізноманіття ²	екстремальні погодні явища ²	водна криза (водозабезпечення) ⁴

Прим: ¹ – економічні; ² – екологічні; ³ – геополітичні; ⁴ – соціальні; ⁵ – технологічні

Дані табл. 1 свідчать, що екологічні ризики, які пов'язані із зміною клімату становить пряму й далекосяжну загрозу сталого розвитку глобального світу.

Специфічна властивість глобальних ризиків полягає у тому, що глобальні ризики формуються і аналізуються на глобальному рівні, а їх вплив проявляється на макроекономічному рівні конкретної країни, причому ступінь впливу одного і того ж глобального ризику на різні макроекономіки неоднакова. Крім того, в умовах інтенсифікації процесу глобалізації, формується тренд неухильного зростання кількості

глобальних ризиків, ймовірності виникнення і ступеня їх впливу а, також, зростає взаємовплив і взаємодія ризиків, надаючи впливу системний характер. У таких умовах обмежуються можливості окремо взятої країни (держави) у поодиножості контролювати і управляти глобальними ризиками. Вищезазначені специфічні властивості висувають особливі вимоги до побудови систем моніторингу глобальних ризиків і формування бази даних, що дозволяють розробляти ефективні заходи щодо попередження та управління ризиками. Організація системи моніторингу глобальних ризиків і можливості трансформації їх в ризики сталого розвитку України є необхідною і актуальною задачею.

Як показано на рис. 1 системи моніторингу рівня глобальних ризиків представляє собою інформаційний сервіс, який буде відповідати за видачу збережених статистичних даних. Користувач матиме можливість побачити потрібні дані окремо за кожним роком.

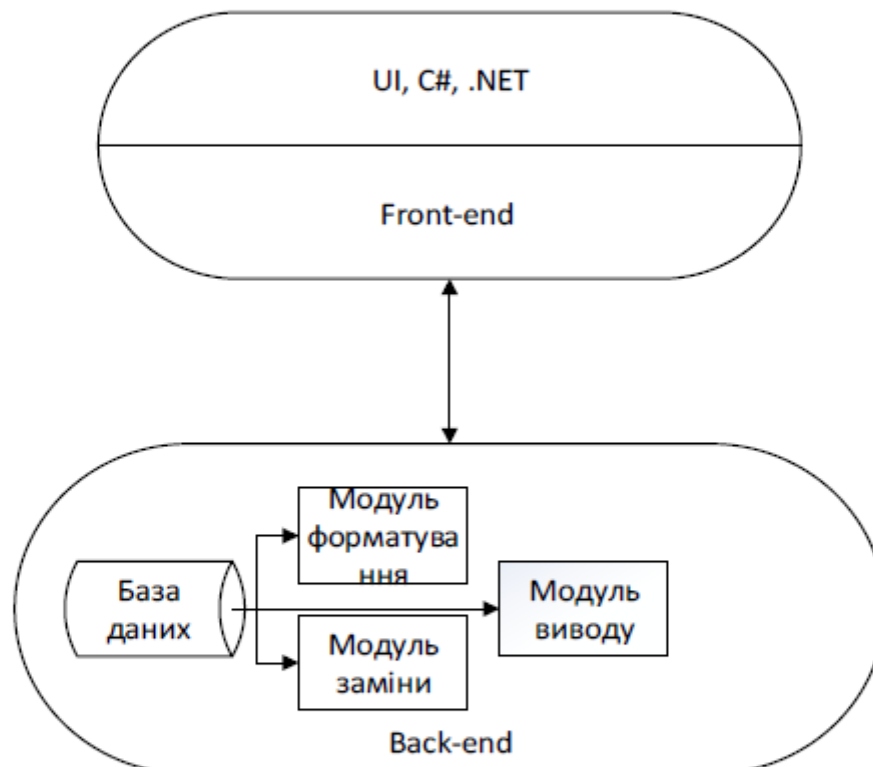


Рисунок 1 - Схема взаємодії компонентів інформаційно-аналітичного додатку

У якості бази даних даної системи буде використано MS SQL Server та Visual Studio 2019 для роботи з нею. Інтерфейс користувача у такому випадку може бути будь-яким, проте розроблений буде найдоцільніший варіант – Desktop-додаток для його реалізації буде використана мова C#, .NET Framework та WinForms UI.

Впровадження такої системи дозволяє враховувати глобальні ризики при розробці стратегічних напрямів забезпечення сталого розвитку країни.

Перелік посилань:

1. World Economic Forum. The Global Risks Report 2020. URL: <https://www.weforum.org/reports/the-global-risks-report-2020>. (дата звернення 02.03.2021)