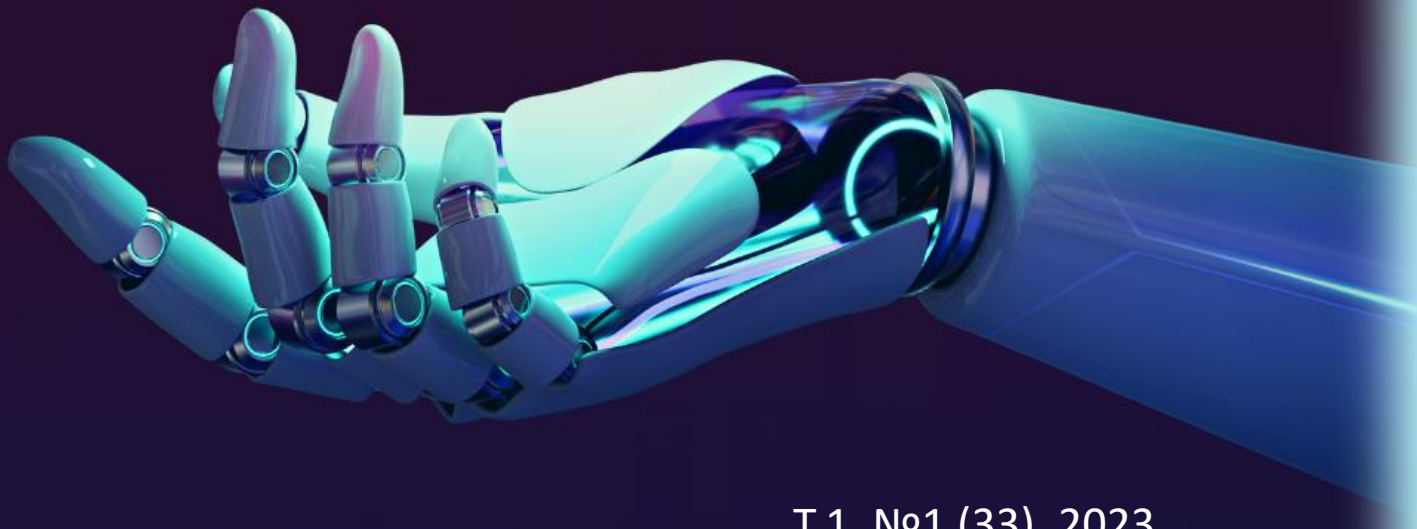
ISSN 2075-4272 (Print),
2786-9024 (Online)

Наукові праці Донецького національного технічного університету. Серія: «Обчислювальна техніка та автоматизація»



Т.1, №1 (33), 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАУКОВІ ПРАЦІ ДОНЕЦЬКОГО НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ

Серія: «Обчислювальна техніка та автоматизація»

Всеукраїнський науковий збірник

Заснований у липні 1998 року

Виходить 1-2 рази на рік

Т1, № 1(33)'2023

Луцьк

2023

УДК 681.5: 658.5: 621.3

Друкується за рішенням Вченої ради Державного вищого навчального закладу «Донецький національний технічний університет» (протокол №4 від 28.03.2024 р.).

У збірнику опубліковано статті науковців, аспірантів, магістрів та інженерів провідних підприємств і закладів вищої освіти України, у яких наведено результати наукових досліджень та розробок, виконаних у 2023-2024 рр. відповідно до напрямків: автоматизація технологічних процесів, інформаційна безпека, інформаційно-вимірювальні системи, електронні та мікропроцесорні прилади, інформаційні технології, кібербезпека та захист критичної інфраструктури, математичне та комп'ютерне моделювання, телекомунікаційні системи та мережі.

Матеріали збірника призначено для викладачів, наукових співробітників, інженерно-технічних працівників, аспірантів і студентів, які досліджують питання інформаційної безпеки, розробки та впровадження інформаційних систем та технологій, розробки й використання автоматичних, інформаційних та електронних систем.

Засновник та видавець – Донецький національний технічний університет.

Ідентифікатор медіа R30-02474 відповідно з додатком до Рішення НРУ з питань телебачення і радіомовлення №139 від 18.01.2024 р.

Збірник включено до списку друкованих (електронних) періодичних наукових фахових видань України, у яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора й кандидата наук (спеціальності 151, 152, наказ Міністерства освіти і науки України № 886 від 02.07.2020 р.).

ISSN 2075-4272 (Print),

ISSN 2786-9024 (Online)

ЗМІСТ

РОЗДІЛ 1. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

РОЗДІЛ 2. КІБЕРБЕЗПЕКА ТА ЗАХИСТ КРИТИЧНОЇ ІНФРАСТРУКТУРИ

РОЗДІЛ 3. ІНФОРМАЦІЙНА БЕЗПЕКА

РОЗДІЛ 4. ЗАГАЛЬНІ ПОНЯТТЯ ІНФОРМАТИКИ

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451440

ВИКОРИСТАННЯ ЗАСОБІВ MACHINE LEARNING І БІБЛІОТЕК PYTHON ДЛЯ ПРОГНОЗУВАННЯ ІЗ ЗАСТОСУВАННЯМ РЕГРЕСІЙ

Я.Л. Байтельман[1], Г.О. Теличко[1], Д.О. Жуковська[1]

[1] Department of Automation and Telecommunications, Donetsk National Technical University, Lutsk, Ukraine

E-mail: hanna.telychko@donntu.edu.ua

Отримано 19.12.2023
Прийнято до публікації 31.12.2023
Опубліковано 01.04.2024

АНОТАЦІЯ

Мета роботи полягає в прогнозуванні середньої суми чеку в залежності від часу перебування покупців в торговельному залі та прогнозуванні середньої суми чеку в залежності від сезонних особливостей попиту з урахуванням постійного щорічного зростання цін.

Проаналізовано методики з прогнозування. Розглянуто використання Python-бібліотек. Розроблено код розв'язання зазначених вище задач з прогнозування через лінійну, поліноміальну регресії та "випадковий ліс", а саме, пошук залежності середньої суми чеку від часу, проведеному покупцями в торговельному залі, та залежності середньої суми чеку від сезонних коливань цін протягом кількох років, і відповідного прогнозування. Здійснено порівняння результатів лінійної та поліноміальної регресії за умови меншого або більшого обсягу вхідних даних, надано пояснення щодо доцільності вибору тої чи іншої при моделюванні нециклічних процесів, і чому для циклічних процесів адекватні результати дає модель "випадкового лісу". Програмний код і приклади вхідних даних викладено у відкритий доступ.

Наукова новизна полягає в шляхах застосування регресій для моделювання та прогнозування середньої суми чеку в залежності від часу перебування покупців в торговельному залі, а також середньої суми чеку в залежності від сезонних особливостей попиту з урахуванням постійного зростання цін.

Обрані приклади циклічного і нециклічного економічних явищ є найпоширенішими, фахівці-практики щоденно стикаються із аналогічними завданнями, тому результати даної роботи надають їм готові рішення, що потребують мінімальної адаптації під практичні потреби, а також демонструють доступність їхнього розгортання на хмарному середовищі AWS і відповідно можливості інтеграції з різними джерелами даних та іншими інформаційними системами.

***Ключові слова**: машинне навчання, регресія, прогнозування, поліноміальна регресія, випадковий ліс, пайтон*

 https://doi.org/10.5281/zenodo.10451440

ВСТУП

Протягом останнього десятиліття спостерігаємо постійно зростаюче взаємне проникнення методів та інструментів, традиційно притаманних одним галузям господарства і відповідним наукам, до інших; з’являються перетини таких дисциплін, які не існували раніше або існували дуже обмежено. Якщо поєднання економічних і комп’ютерних наук розпочалося майже одразу, як тільки з’ясувалося, що обчислювальні потужності можуть вирішувати питання автоматизації фінансових розрахунків, то засоби машинного навчання є відносно новими для практичного використання в сфері економіки, особливо з метою прогнозування і тим більше з урахуванням специфіки підготовки фахівців. Доволі часто, економісти-практики мають обмежене уявлення про інструменти моделювання і прогнозування засобами машинного навчання, або вважають, що такі засоби не є для них доступними через складність використання. Проте в сучасному світі міждисциплінарні зв’язки стають все ціннішими, постійне самостійне підвищення кваліфікації, в тому числі у споріднених сферах, розглядається, як запорука конкурентоспроможності фахівців, а необхідність розв’язання прикладних задач тільки зростає.

АНАЛІЗ ЛІТЕРАТУРНИХ ДАНИХ ТА ПОСТАНОВКА ПРОБЛЕМИ

В освітній програмі “Економічна аналітика” є така обов’язкова дисципліна, як “Соціально-економічне прогнозування та проектування”, один із наявних навчальних посібників – “Прогнозування соціально-економічних процесів” 2021 року видання [1]. Цей посібник пропонує фундаментальну теоретичну базу із детальним описом методів, наводить пояснення математичного апарату, включаючи розгляд регресійних моделей, проте робочим інструментом обрано Microsoft Excel, що є цілком виправданим для опанування знаннями та навичками, для практичного розв’язання певних розрахункових завдань, але не в автоматизованому режимі. Питання втілення певного автоматизованого рішення з можливістю інтегрувати в інші системи, навіть такі прості, як веб-сайт, неможливо вирішити засобами Microsoft Excel, тоді як після закінчення навчання перед вчорашніми студентами – сьогоднішніми молодими співробітниками комерційних компаній або державних установ цілком вірогідно в найближчому майбутньому поставатимуть конкретні робочі задачі, які потребуватимуть масштабування та інтеграції. В освітній програмі бакалаврів “Економічна кібернетика” є значно складніший за викладенням матеріалу навчальний посібник “Прогнозування соціально-економічних процесів” 2022 року видання [2], адже спеціалізація цих фахівців є більш вузько направленою. Проте виникає питання щодо самоосвіти фахівців, які не мають можливості через обмеження в часі опанувати освітню програму в повному обсязі, і яким водночас необхідно вирішувати практичні задачі. В освітній програмі “Економічна аналітика та бізнес-статистика” є курс з основ програмування на мові Python [3], що вже значно ближче до широкого спектру прикладних потреб, але знову ця інформація доступна в межах формальної академічної освіти і не охоплює людей, зайнятих на постійній основі у виробництві або сфері послуг, а також підприємців – засновників стартапів. Для останньої групи критичним є здатність відшукати потрібні матеріали і в стислий термін (тижні краще ніж місяці) оволодіти новими для них інструментами, перевірити, чи вони підходять для вирішення їхніх задач, одночасно такі інструменти мають відповідати вимогам масштабування та інтеграції з іншими інформаційними системами, програмним забезпеченням, тощо. В англомовному середовищі є досить багато змістовних робіт, як наукового так і прикладного характеру, з питань прогнозування взагалі [4 – 6], так і економічних явищ зокрема, проте робота з ними потребує високого рівня володіння англійською мовою.

Формулювання проблеми: в умовах постійно зростаючої потреби в міждисциплінарних компетенціях існує ряд прикладних задач з прогнозування, зокрема, економічного, серед яких визначення бажаного часу перебування покупців в торговельному залі для отримання найбільшої середньої суми чеку, а також визначення залежності сезонних коливань середньої суми чеку із урахуванням щорічного зростання цін, розв’язання яких можливе засобами машинного навчання.

Мета: розробка програмного коду на основі бібліотек Python для прогнозування зазначених вище явищ. Вибір мови продиктований її поширеністю, доступністю до вільного і безкоштовного використання навіть в комерційних цілях [7], відносною легкістю опанування, можливістю виконувати досить складні вправи з програмування без необхідності встановлення платного або складного середовища.

Задачі:

1. Аналіз тематичних наукових джерел і методичних матеріалів.
2. Експериментальна перевірка запропонованих в них прикладів, їхня адаптація під зазначені вище завдання з прогнозування.

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451440

3. Розробка і тестування програмного коду моделей прогнозування.

4. Аналіз результатів, отриманих від розроблених моделей.

МАТЕРІАЛИ ТА МЕТОДИ ДОСЛІДЖЕНЬ

В роботі використано методи структурного і порівняльного аналізу, з інформаційним і аналітичним підходом розглянуто наукову та методичну літературу, а також онлайн ресурси, застосовано експериментальні методи для перевірки програмного коду.

В середовищі розробників програмного забезпечення існує відомий ресурс для опанування основами різних мов програмування W3School [8]. Його популярність зумовлена тим, що він дозволяє покроково знайомитися з синтаксисом, специфікою, вбудованими засобами та додатковими бібліотеками різних мов, значно скорочуючи шлях навчання для осіб, які вже мають уявлення про теоретичні основи програмування, як-то розуміють принципи об'єктно-орієнтованого підходу, знайомі зі структурами даних, тощо, тому для них вивчення нової мови фактично зводиться до пошуку відповіді на питання, як записати засобами цієї мови те, що вони вже вміють іншою мовою. Одночасно, для новачків покрокове подання матеріалу разом із прикладами і навіть симулятором, де можна подивитись на результати виконання прикладу без розгортання середовища програмування, дає можливість вже через 1–2 години самостійної роботи отримати функціонуючий фрагмент коду, готового для подальших експериментів. W3School доступний англійською мовою, були спроби перекласти його на українську [9], але саме розділи про прогнозування досі не перекладено (на момент здійснення даної роботи, жовтень 2023).

В процесі розгляду публікацій за темою дослідження виникли певні питання до терміну "прогнозування" (англійською prediction), можливо через ось це трактування: "The term regression is used when you try to find the relationship between variables. In Machine Learning, and in statistical modeling, that relationship is used to predict the outcome of future events." [10]. В перекладі українською: "Термін регресія використовується коли ви намагаєтесь знайти зв'язок між змінними. В машинному навчанні і в статистичному моделювання такий зв'язок використовується для прогнозування результатів майбутніх подій." Тлумачний словник англійської мови дає таке визначення слову predict: "say or estimate that (a specified thing) will happen in the future or will be a consequence of something". В перекладі українською: "говорити чи оцінювати, що (певна річ) трапиться в майбутньому, або стане наслідком чогось". Походження цього слова в англійській мові, вважається, відбулось в ранньому 17-му столітті від латинського praedict – "зробити відомим заздалегідь, оголосити", від дієслова praedicere, від prae- "перед" + dicere "говорити". Українське слово "прогноз", так як і англійське "prognosis", походить від грецького prognōsis, від pro- " перед" + gignōskein "знати". На побутовому рівні немає сумнівів, що слова "прогноз", "prediction" мають явне відношення до визначення невідомого, яке лежить саме в майбутньому, того, що ще не сталося. Однак, далі в даній роботі з'ясовується, що питання термінології є більш складним, та інколи призводить до хибних трактувань.

В даній роботі використано матеріали із розділу Machine Learning ресурсу W3 School, а також матеріали детального опису бібліотеки Skforecast [11]. Для виконання коду Python обрано ресурси хмарного середовища AWS Amazon, а саме, розгорнуто найменший доступний EC2 сервер із операційною системою Linux, здійснено оновлення Python до новішої доступної версії, встановлено додаткові бібліотеки і пакети. Доступ до серверу здійснюється в терміналі, через канал ssh, для роботи з файлами, включаючи редагування, використовувався Midnight Commander [12]. Такий набір інструментів обумовлений можливістю швидкого розгортання. Детальні покрокові інструкції наведено в документації AWS Amazon [13-14]. Варто зауважити, що нові користувачі, які вперше зареєструвалися на AWS Amazon, мають право на безкоштовний набір певних мінімальних сервісів протягом року, відомий як Free Tier, а при створенні EC2 серверу є позначки, які саме типи підпадають під умови безкоштовного користування в рамках Free Tier. Повний код Python для кожного прикладу доступний на інтернет ресурсі автора, далі в тексті вказуються назви файлів, які можна завантажити із зазначеного ресурсу [15].

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Торговельні мережі застосовують особливі підходи, що спонукають відвідувачів до так званих "імпульсних" покупок, тобто не завжди покупці придбають лише необхідне або заздалегідь визначене за списком покупок, з іншого боку, покупці з довгим списком, вочевидь, проводять більше часу в магазині. Існує певна залежність між часом, проведеним в торговельному залі, і сумою, яка витрачається на покупки. Для аналізу даної залежності використовується поняття "середня сума" чеку за деякий часовий інтервал. В якості вхідних даних для

 https://doi.org/10.5281/zenodo.10451440

прогнозування обрано масив таких усереднених значень (Таблиця 1). В даному випадку це довільно обраний масив чисел, що зростають. В реальності такі значення отримуються через спостереження за покупцями та аналізом інформації щодо сум їхніх покупок. Для прикладів даного дослідження достовірність цих даних не є важливою, важливим є лише направлення зміни (зростання з часом).

В процесі дослідження було проведено порівняння моделей на основі лінійної та поліноміальної регресій. Для поліноміальної регресії використано функцію linregress із бібліотеки scipy (Рисунок 1). Повний код наведено в файлі `linear.py` включно із прикладом побудування графіків. Слово "регресія" означає "спрощення", тобто вся сукупність вхідних даних "спрощується" до певної простої залежності. Сутність лінійної регресії зводиться до автоматичного пошуку закономірності, яка може бути описана лінійним рівнянням, а в графічному представленні має вигляд прямої.

В процесі дослідження було проведено порівняння моделей на основі лінійної та поліноміальної регресій. Для поліноміальної регресії використано функцію linregress із бібліотеки scipy (Рис. 1). Повний код наведено в файлі `linear.py` включно із прикладом побудування графіків. Слово "регресія" означає "спрощення", тобто вся сукупність вхідних даних "спрощується" до певної простої залежності. Сутність лінійної регресії зводиться до автоматичного пошуку закономірності, яка може бути описана лінійним рівнянням, а в графічному представленні має вигляд прямої.

```
import matplotlib.pyplot as plt
from scipy import stats
time = [5,10,15,20,25,30,35,40,45,50,55,60]
money = [5,7,25,42,88,91,103,150,152,190,195,200]
slope, intercept, r, p, std_err = stats.linregress(time, money)
def myfunc(time):
    return slope * time + intercept

model = list(map(myfunc, time))
print("the coefficient of correlation = "+ str(r))
```

Рис. 1. Лінійна регресія

Для наведених вище даних отримано коефіцієнт кореляції 0.98, що свідчить про адекватну відповідність залежності, але судячи з графіку на Рисунку 2, пряма лінія не охоплює хвилини 25, 35, 40, 50, 60, для їхнього включення потрібна крива, тому далі розглядається поліноміальна регресія і перевіряється, чи вона краще охоплює всі задані точки. Поліноміальна регресія – це пошук більш складної закономірності, яка описується рівнянням із змінною, що підносяться до другого, третього або більших ступенів, а в графічному представленні схожа на дугу.

Таблиця 1. Залежність суми чеку від часу

Час (хвилини)	5	10	15	20	25	30	35	40	45	50	55	60
Сума ($)	5	7	25	42	88	91	103	150	152	190	195	200

Для наведених вище даних отримано коефіцієнт кореляції 0.98, що свідчить про адекватну відповідність залежності, але судячи з графіку на Рис. 2, пряма лінія не охоплює хвилини 25, 35, 40, 50, 60, для їхнього включення потрібна крива, тому далі розглядається поліноміальна регресія і перевіряється, чи вона краще охоплює всі задані точки. Поліноміальна регресія – це пошук більш складної закономірності, яка описується рівнянням із змінною, що підносяться до другого, третього або більших ступенів, а в графічному представленні схожа на дугу.

Повний код побудови моделі поліноміальної регресії із застосуванням функцій `poly1d` і `polyfit` з бібліотеки `numpy` та функції `r2_score` для визначення коефіцієнту детермінації з бібліотеки `sklearn` міститься в файлі `polynom.py`.

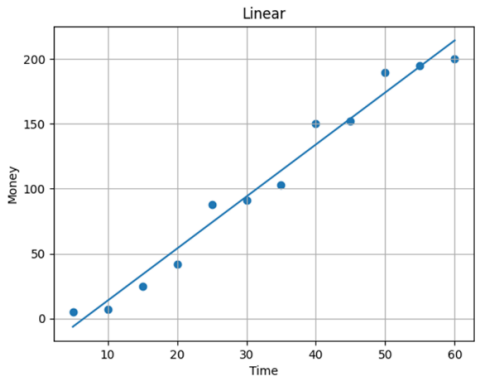


Рис. 2. Графік лінійної регресії

Спочатку здійснюється тренування моделі на відомих значеннях часу і сум чеків, далі – прогнозування через передачу до моделі значень часу, для яких сума чеку не є відомою. Виконання дає коефіцієнт детермінації рівний

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451440

0.98, що вказує на високу надійність результатів прогнозу. Спочатку модель використовується для заповнення пропущених значень часу (7, 12, 18, 33, 51 хвилин). Результати прогнозу графічно представлені на Рис. 3, блакитні точки відповідають відомим значенням, а помаранчеві показують результати прогнозу, крива лінія є графіком поліноміальної моделі.

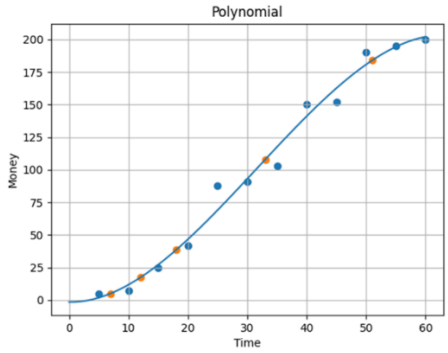


Рис. 3. Графік поліноміальної регресії і прогнозу в межах заданого діапазону

Прогнозування за межами початкового часового діапазону виконується тим самим кодом моделі в файлі polynom.py, але з підставленням нових значень часу (65, 70, 75, 90, 120 хвилин). Результати прогнозування усереднених витрат відповідно до даних часу, що лежать за межами вхідної інформації, наведено на Рис. 4.

Якщо “близькі” значення 65 і 75 хвилин дають якісь, можливо, адекватні результати, то зі збільшенням часового інтервалу, відходячи все далі від межі початкового діапазону спостерігається стрімке падіння в напрямку від’ємних величин.

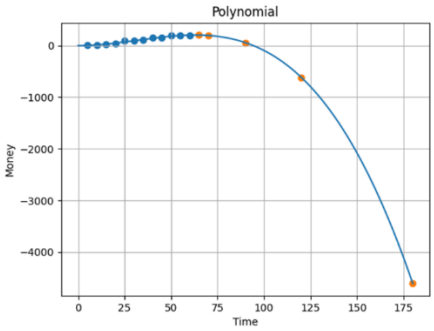


Рис. 4. Графік поліноміальної регресії і прогнозу за межами заданого діапазону

Таке прогнозування не має ніякого сенсу, тож висновок – поліноміальна регресія дозволяє більш або менш акуратно доповнювати пропущені дані в межах заданого діапазону, а також є корисною для графічного представлення, або побудови кривої саме шляхом доповнення точок на графіку, для яких не існувало вхідних даних. Не складно уявити, що при застосуванні лінійної регресії, нові прогнозовані точки лежатимуть на продовжені прямої, але в рамках прикладу даного дослідження їм так само бракуватиме сенсу, адже немає сумнівів, що кількість грошей в кишенях покупців є величиною кінцевою і навіть якщо вони проведуть цілий день в магазині, то не можуть витратити більше, ніж мають. В якості експерименту було здійснено виконання прогнозу на тій самій моделі, але із збільшенням обсягу початкових даних (Рис. 5).

Збільшення кількості елементів в масивах вхідних значень не призвело до покращення екстраполяції за межами вхідного часового діапазону, принаймні в рамках розгляду залежності суми чеку від часу, проведеному в супермаркеті. Зі збільшенням відходження від меж заданого для тренування моделі діапазону адекватність прогнозування втрачається, що є очікуваним для прогнозування залежності між обмеженими величинами, в даному випадку це гроші у кожного покупця і час, який вони можуть провести в магазині.

При розгляді залежності середньої суми чеку від місяця протягом кількох років, йдеться про процес з певними коливаннями, а прогноз має давати уявлення, по-перше, про зміну витрат відповідно до циклічних подій, таких як свята, сезон літніх відпусток, початок навчального року, і по-друге, про щорічне зростання цін, викликане інфляцією.

```
time =  [5, 5,  6,  8,  9,  9,  10, 11, 13,  14,  14, 15, 15,  16, 17, 19,   20,
 21, 22, 25, 25, 25, 26, 28, 30, 32, 34,  35,  37,  38,  39,  40, 40,  41,  43,
  45, 49,  50, 52,  55,  57,  57,  60]
money = [5, 5.3,5.8,6.4,6.2,6.9, 7, 6.8, 8.5,19.8,17, 25, 28.5,24, 39, 32.5, 42,
 49, 65, 99, 79, 88, 90, 103,91, 99, 100, 103, 140, 135, 160, 150,145, 155, 160,
  152,160, 190,195, 198, 206, 210, 202 ]

model = numpy.poly1d(numpy.polyfit(time, money, 3))
time_line = numpy.linspace(0, 180, 180)
plt.scatter(time,money)
plt.plot(time_line, model(time_line))

print ("Filling the gaps:")
predicted_time = [7,12,18,33,51]
predicted_money =[]
for t in predicted_time:
    predicted_money.append(model (t))
plt.scatter(predicted_time, predicted_money)

print ("Prediction of unknown:")
predicted_time2 = [65,70,90,120,180]
predicted_money2 =[]
for t in predicted_time2:
    predicted_money2.append(model (t))
plt.scatter(predicted_time2, predicted_money2)
```

Рис. 5. Поліноміальна регресія

Для перевірки цього припущення був підготовлений масив даних, в якому кожний елемент включає дату та середню суму чеку, всього 6 значень на місяць для 4 послідовних років. Повний набір міститься у файлі `cyclic_data.csv`. Була розроблена модель із застосуванням бібліотеки `sklearn` для прогнозування через алгоритм випадкового лісу (Рис. 6). Повний код наведено у файлі `predict.py`. Без глибокого занурення в цей алгоритм слід пояснити принцип його дії: для класифікації (в даному випадку, для визначення сезонних

 https://doi.org/10.5281/zenodo.10451440

перепадів) будуються логічні дерева прийняття рішень, кожне розгалуження – це певна умова (в даному випадку, чи закінчується тренд на спадання чи зростання).

```
regressor = RandomForestRegressor(max_depth=5, n_estimators=30, random_state=123)
# max depth of each tree, number of trees, random seed
forecaster = ForecasterAutoreg(regressor = regressor,lags = 16)
#lags - observations
forecaster.fit(y=data['money'])
predictions = forecaster.predict(12) #months
```

Рис. 6. Регресія "випадкового лісу"

Необхідність експериментального визначення параметрів максимальної глибини дерева (кількість рівнів розгалужень), кількості дерев, а також кількості спостережень (обчислень) для отримання найкращих результатів прогнозування варта окремого зауваження; в ході дослідження було здійснено близько 50 експериментів для отримання кінцевих значень, які привели до найкращих результатів прогнозу при найменшому часі виконання коду. Також важливо звернути увагу на існування дещо складних для початківців методів перевірки ефективності моделі, таких як зворотне тестування, проте метою даної роботи є найпростіша реалізація прогнозування, тому навіть фрагмент коду для тестування моделі після її тренування було виключено.

На Рис. 7 надається графічне представлення

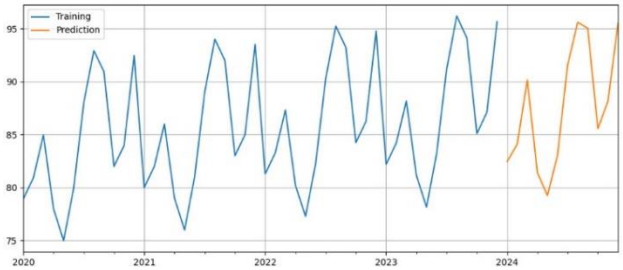


Рис. 7. Графік циклічних подій

відомих даних для 4 послідовних років, які було використано для тренування моделі (блакитним), і прогнозу на 5-й рік (помаранчевим).

Визначено певні сезонні зростання і падіння – найвищі щорічні середні значення припадають на середину літа, що відповідає періоду відпусток, а також на період перед різдвяними та новорічними святами. Також очевидне щорічне зростання всіх значень, так найвища сума для 2020 року була в районі 93, в 2021 наблизилася до 95, в 2022 дещо перевищила 95, а в 2023 значно перевищила 95. Аналогічні щорічні зростання спостерігаються і на інших екстремумах, що вказує на постійне зростання цін, ймовірно, внаслідок інфляції. Очікувано, прогноз на 2024 рік зберіг зразок, в результаті тренування із застосуванням даних попередніх років модель вірно визначила сезонні коливання і відповідно показала також зростання на більшості екстремумів окрім лише двох "піків", які виявилися дещо нижчими за такі самі місяці попереднього року. В принципі, результати прогнозу можна вважати задовільними, так як модель визначила загальні тренди щорічного зростання і сезонних коливань. Безсумнівно, феномени інфляції і росту цін є значно складнішими, простий пошук закономірності через порівняння історичних величин не враховує повною мірою широкий спектр різних чинників, тому запропонований приклад слід розглядати умовно, в якості демонстрації застосування методу випадкового лісу для прогнозування циклічних явищ.

ОБГОВОРЕННЯ РЕЗУЛЬТАТІВ

Обговорення результатів мали місце під час занять з курсів "Моделювання та оптимізація систем управління" та "Інтелектуальні технології керування" програми підготовки магістрів групи СУАм-23 Донецького національного технічного університету.

ВИСНОВКИ

1. Проведено аналіз джерел, запропонований в них матеріал адаптовано під вимоги даної роботи.
2. Розроблено моделі прогнозування середньої суми чеку в залежності від проведеного в магазині часу, які демонструють, що лінійна і поліноміальна регресія, навіть за умови збільшення обсягу вхідних даних, не дають адекватного прогнозу поза межами заданого діапазону, проте такі регресії допомагають відновити пропущені значення в його межах, а також є корисними для візуалізації даних.
3. Розроблено модель для прогнозування залежності середньої суми чеку від сезонних коливань в умовах зростання цін на основі даних попередніх періодів із застосуванням методу випадкового лісу.
4. Всі приклади виконання прогнозування супроводжуються поясненнями та графіками.
5. Зазначені вище приклади коду для відносно простих задач з прогнозування на мові Python із застосуванням бібліотек `numpy` і `sklearn` не є складним навіть для початківців, і тому наведені приклади можуть бути в нагоді для практичного використання із внесенням мінімальних змін. Також надано рекомендації щодо доступного хмарного середовища виконання коду.

ЛІТЕРАТУРА

[1] М.П. Галущак., О.Я. Галущак, Т.І. Кужда, "Прогнозування соціально-економічних процесів: навчальний посібник для економічних спеціальностей".

ISSN 2786-9024

Тернопіль, 2021. [Онлайн]. URL: http://surl.li/rlbjn. Дата звернення: 10.11.2023.
[2] О. А. Жуковська, "Прогнозування соціально-економічних процесів: комп'ютерний практикум: навч. посіб. для здобувачів ступеня бакалавра за освітньою програмою "Економічна кібернетика" спеціальності 051 Економіка". Київ: КПІ ім. Ігоря Сікорського, 2022. [Онлайн]. URL: http://surl.li/rlblj. Дата звернення: 10.11.2023.
[3] О.М.Вільчинська, "Силабус з навчальної дисципліни "Основи програмування Python". Львів: Львівський національний університет ім. Івана Франка, економічний факультет, кафедра статистики, 2022. [Онлайн]. URL: http://surl.li/rlbne. Дата звернення: 10.11.2023.
[4] M.Bowles, "Machine learning in Python: essential techniques for predictive analysis". John Wiley & Sons, 2015.
[5] M.Peixeiro, "Time series forecasting in python". Simon and Schuster, 2022.
[6] J.Yoon, "Forecasting of Real GDP Growth Using Machine Learning Models: Gradient Boosting and Random Forest Approach", *Comput Econ*, vol. 57, 247–265, 2021, doi: 10.1007/s10614-020-10054-w.
[7] Python License, Version 2. [Онлайн]. URL: http://surl.li/rlbpi. Дата звернення: 10.11.2023.
[8] W3School. [Онлайн]. URL: https://ww.w3schools.com. Дата звернення: 10.11.2023.
[9] W3School Українською. [Онлайн]. URL: http://surl.li/rlbqo. Дата звернення: 10.11.2023.
[10] W3School. Machine Learning - Linear Regression. [Онлайн]. URL: http://surl.li/rlbrd. Дата звернення: 10.11.2023.
[11] A.R.Joaquín, J.E.O.Rodrigo. Skforecast: time series forecasting with Python and Scikit-learn. [Онлайн]. URL: http://surl.li/rlbrz. Дата звернення: 10.11.2023.
[12] Midnight Commander. [Онлайн]. URL: http://surl.li/rlbuy. Дата звернення: 10.11.2023.
[13] Set up to use Amazon EC2. [Онлайн]. URL: http://surl.li/rlbvn. Дата звернення: 10.11.2023.
[14] Install Python, pip, and the EB CLI on Linux. [Онлайн]. URL: http://surl.li/rlbwp. Дата звернення: 10.11.2023.
[15] Я.Л. Байтельман. Код Python і приклади даних для прогнозування. [Онлайн]. URL: http://surl.li/rlbxq. Дата звернення: 10.11.2023.

USE OF MACHINE LEARNING TOOLS AND PYTHON LIBRARIES FOR PREDICTION THROUGH REGRESSIONS

Yakiv (Jacob) Baytelman, Hanna Telychko, Daria Zhukovska

The purpose of this work is forecasting the average check amount depending on the time spent by customers in the retail area and forecasting the average check amount based on seasonal demand characteristics considering the constant annual price growth.

Forecast methods were analysed. The use of Python libraries was considered. The code was developed for solving the above-mentioned forecasting tasks through linear, polynomial and random forest regressions, specifically, the search for the dependence of the average check amount on the time spent by customers in the retail area and the dependence of the average check amount on seasonal price fluctuations over several years. A comparison of the results of linear and polynomial regression was made under conditions of smaller or larger volumes of the input data, explanations were provided regarding the appropriateness of choosing one or the other when modelling non-cyclical processes and why the random forest model provides adequate results for cyclical processes. The source code and examples of input data were shared for public access.

Scientific novelty lies in the ways of applying regressions for modelling and forecasting the average check amount depending on the time customers spend in the retail area as well as the average check amount depending on seasonal demand characteristics with consideration of constant price growth.

The selected examples of cyclical and non-cyclical economic phenomena are the most common ones, and practitioners face similar tasks daily. Therefore, the results of this work provide them with ready-made solutions that require minimal adaptation to practical needs. Additionally, it demonstrates the feasibility of deploying them in the AWS cloud environment and the potential for integration with various data sources and other information systems.

***Keywords**: machine learning, regression, prediction, polynomial regression, random forest, python.*

REFERENCES

[1] M.P. Halushchak, O.Ia. Halushchak, T.I. Kuzhda, "Prohnozuvannia sotsialno-ekonomichnykh protsesiv: navchalnyi posibnyk dlia ekonomichnykh spetsialnostei". Ternopil, 2021. [Online]. URL: http://surl.li/rlbjn. Accessed: 10.11.2023. (In Ukrainian).
[2] O. A. Zhukovska, "Prohnozuvannia sotsialno-ekonomichnykh protsesiv: kompiuternyi praktykum: navch. posib. dlia zdobuvachiv stupenia bakalavra za osvitnoiu prohramoiu "Ekonomichna kibernetyka" spetsialnosti 051 Ekonomika". Kyiv: KPI im. Ihoria Sikorskoho. Kyiv, 2022. [Online]. URL: http://surl.li/rlblj. Accessed: 10.11.2023. (In Ukrainian).
[3] O.M.Vylchinska, "Sylabus z navchalnoi dystsypliny "Osnovy prohramuvannia Python". Lviv: Lviv national university im. Ivana Franka, department of economics and statistics, 2022. [Online]. URL: http://surl.li/rlbne. Accessed: 10.11.2023. (In Ukrainian).
[4] M.Bowles, "Machine learning in Python: essential techniques for predictive analysis". John Wiley & Sons, 2015.

[5] M.Peixeiro, “Time series forecasting in python”. Simon and Schuster, 2022.
[6] J.Yoon, “Forecasting of Real GDP Growth Using Machine Learning Models: Gradient Boosting and Random Forest Approach”, Comput Econ, vol. 57, 247–265, 2021, doi: 10.1007/s10614-020-10054-w.
[7] Python License, Version 2. [Online]. URL: http://surl.li/rlbpi. Accessed: 10.11.2023.
[8] W3School. [Online]. URL: https://ww.w3schools.com. Accessed: 10.11.2023.
[9] W3School In Ukrainian. [Online]. URL: http://surl.li/rlbqo. Accessed: 10.11.2023. (In Ukranian).
[10] W3School. Machine Learning - Linear Regression.. [Online]. URL: http://surl.li/rlbrd. Accessed: 10.11.2023.
[11] A.R.Joaquín, J.E.O.Rodrigo. Skforecast: time series forecasting with Python and Scikit-learn. [Online]. URL: http://surl.li/rlbrz. Accessed: 10.11.2023.
[12] Midnight Commander.. [Online]. URL: http://surl.li/rlbuy. Accessed: 10.11.2023.
[13] Set up to use Amazon EC2. [Online]. URL: http://surl.li/rlbvn. Accessed: 10.11.2023.
[14] Install Python, pip, and the EB CLI on Linux. [Online]. http://surl.li/rlbwp. Accessed: 10.11.2023.
[15] J.L.Baytelman. Kod Python i pryklady danyh dlia prognozuvanyia. [Online]. URL: http://surl.li/rlbxq. Accessed: 10.11.2023. (In Ukranian).

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451666

ПРІОРИТЕЗАЦІЯ ВИМОГ ПРИ РОЗРОБЦІ ПРОЕКТІВ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОБ'ЄКТІВ КРИТИЧНОЇ ІНФРАСТРУКТУРИ

Я.Ю. Дорогий[1], О.О. Дорога-Іванюк[2]

[1] Department of Applied Mathematics and Informatics, Donetsk National Technical University, Lutsk, Ukraine
[2] Department of Computer Science and Software Engineering, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine

E-mail: yaroslav.dorohyi@donntu.edu.ua

Отримано 30.10.2023
Прийнято до публікації 09.11.2023
Опубліковано 01.04.2024

АНОТАЦІЯ

Мета роботи полягає в розробці алгоритму пріоритезації вимог при розроблюванні програмного забезпечення для проектів об'єктів критичної інфраструктури.

Розробка вимог є базовою фазою будь-якого проекту програмного забезпечення, оскільки ця фаза пов'язана з ідентифікацією вимог, обробкою та маніпулюванням. Основним джерелом цих вимог є зацікавлені сторони проекту з урахуванням обмежень та лімітів проекту. Кількість вимог різна для кожного проекту програмного забезпечення для об'єкту критичної інфраструктури, тому термін пріоритезації вимог стосується визначення пріоритетності порядку виконання вимог до програмного забезпечення відповідно до міркувань і рішень зацікавлених сторін. Для вирішення оптимізаційних задач використовуються різні запропоновані алгоритми оптимізації. В даній роботі наведено основні етапи базових алгоритмів оптимізації, деякі їх модифікації з метою покращення їх ефективності у розв'язанні такого типу задач. Також, у цій статті пропонується гібридний підхід, заснований на алгоритмах оптимізації WOA та GWO, утворений шляхом поєднання переваг кожного алгоритму з метою визначення пріоритетності вимог до програмного забезпечення ОКІ. Крім того, наведено набір даних з проекту SKUDA, який використано в даному дослідженні та відповідає вимогам реального програмного проекту для оцінки запропонованого методу.

Наукова новизна полягає в модифікації, застосуванні та поєднанні результатів відомих алгоритмів GWO та WOA для розв'язання задачі пріоритезації вимог для проектів програмного забезпечення об'єктів критичної інфраструктури.

Запропонований алгоритм дає точність 92% для запропонованого набору вимог.

***Ключові слова**: пріоритезація вимог, WOA, GWO, об'єкт критичної інфраструктури, ОКІ, гібридний підхід*

 https://doi.org/10.5281/zenodo.10451666

ВСТУП

Інженерія вимог (ІВ) є однією з найважливіших галузей у сфері інженерії програмного забезпечення. Крім того, вона вважається найважливішою фазою у життєвому циклі розробки програмного забезпечення. Ця фаза включає ідентифікацію та виборку вимог, аналіз та перевірку вимог і їх документацію. Практично в усіх проектах програмного забезпечення існують обмеження у процесі розробки, такі як бюджет та час до виходу продукції на ринок, що призводить до поетапної поставки проектів програмного забезпечення. Таким чином, у великих проектах існує більше одного зацікавленого в результатах учасника, що ускладнює процес вибору того, який випуск слід розробляти першим. Ця складність змушує інженерів програмного забезпечення ефективно визначати пріоритети вимог, щоб приймати правильні рішення щодо поставки та розробки проекту [1-2].

Отже, визначення пріоритетів вимог (RP) є найважливішою частиною ІВ, яка входить до стадії аналізу вимог. Визначення пріоритетів вимог вважається однією з найважливіших дій у процесі створення програмного проекту та надання якісної системи, яку потребує замовник. У випадках, коли у проекті існують жорсткий графік виконання, недостатні ресурси та високі очікування споживачів, важливо опублікувати найважливіші характеристики якнайшвидше. Саме це і вимагає визначення пріоритетів вимог.

Процес визначення пріоритетів вимог підвищує участь учасників проекту, включаючи їх у вирішення питань, які вимоги повинні бути включені в програмне забезпечення разом з їхньою важливістю та впливом на процес розробки. Ця участь допомагає учасникам розуміти обмеження та ліміти ресурсів проекту та вирішувати конфлікти між різними точками зору, які дійсно впливають на розробку програмного забезпечення. Ці конфлікти виникають з різних цілей та ролей учасників проекту [3]. Таким чином, у проектах з великою кількістю вимог визначення пріоритетів стає основою успіху чи невдачі програмного продукту відповідно до обмежень та лімітів проекту. Ця участь та взаємодія спрямовані на визначення пріоритетів вимог з урахуванням їх важливості ефективним способом та корисним порядком.

З цієї причини можна використовувати техніку кластеризації для класифікації вимог за їх важливістю. Кластеризація даних - одна з найважливіших функцій у сфері видобутку даних, яка востаннє привертає увагу багатьох авторів, дослідників та експертів. Кластеризація - це неконтрольоване упорядкування типів у групи [4]. Основна мета кластеризації даних полягає в ідеї групування об'єктів у групи на основі схожості та відмінності між ними [5-6].

Загалом, кластеризацію можна поділити на дві основні категорії: жорстку кластеризацію і м'яку кластеризацію. У жорсткій кластеризації об'єкти даних належать лише до одного кластеру, в той час як у м'якій кластеризації всі окремі об'єкти даних належать до окремих класів у певному діапазоні [5, 7]. Основна мета кластеризації даних полягає в тому, що вона оптимально сортує всі N даних у K кластерів так, щоб неявні типи даних стали видимими [8]. Таким чином, кожен кластер містить об'єкти даних, які схожі за характеристиками, а також кластери відрізняються один від одного. Одним із найважливіших методів, які використовуються у процесі дослідження даних, нейрокомп'ютерних технологій, сегментації зображень та інших інженерних задач, є кластерний аналіз [9]. Наразі багато методів кластеризації були запропоновані дослідниками та поділені на алгоритми кластеризації на основі моделі, ієрархічні алгоритми кластеризації, алгоритми кластеризації на основі розбиття, алгоритми кластеризації на основі сіток та алгоритми кластеризації на основі щільності [8, 10]. Дані, які розділяються на K кластерів, використовують відстань Евкліда як міру у алгоритмах кластеризації на основі розбиття, також в алгоритмах кластеризації на основі ієрархії створюється дерево груп.

Останнім часом багато дослідників запропонували безліч метаевристик і евристичних алгоритмів для вирішення проблем, які виникають у зв'язку з складними наборами даних. Проте більшість технік, які запропоновані для вирішення проблем оптимізації, базуються на метаевристичних алгоритмах [9, 11]. Основна мета метаевристи-чних алгоритмів – знайти оптимальні рішення для формування кластерів даних та зменшення проблеми локальних мінімумів [12-13]. Останні метаевристичні алгоритми оптимізації – це алгоритм GWO (Grey Wolf Optimization) та алгоритм WOA (Whale Optimization Algorithm).

Алгоритм GWO був запропонований Мірджалілі та іншими у 2014 році [14]. Цей алгоритм моделює поведінку сірих вовків у природі під час полювання. Сірі вовки є одними з найвідоміших хижаків в природі. Зазвичай вони живуть у групах розміром від 5 до 12 особин. У цих вовків існують стійкі правила в соціальній ієрархії. Згідно з [14], сірі вовки включають альфа, бета, омега та дельта вовків. Альфа-вовк є лідером стаї і відповідає за прийняття рішень щодо полювання та інших діяльностей. Бета-вовк допомагає вищому рівню приймати рішення. Омега-вовки відповідальні за передачу інформації на найвищі рівні. Усі інші вовки у стаї називаються дельтами.

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451666

Алгоритм WOA був запропонований Мірджалілі та Льюїсом у 2016 році [10]. Основна його мета – знайти глобальне оптимальне рішення для будь-якої задачі оптимізації. Головна відмінність цього алгоритму від інших полягає у принципі, який розвиває рішення-кандидата на кожній ітерації оптимізації. Крім того, процес полювання китів за допомогою бульбашкової мережі відображає процес полювання горбатих китів на джерело їжі.

АНАЛІЗ ЛІТЕРАТУРНИХ ДАНИХ ТА ПОСТАНОВКА ПРОБЛЕМИ

Як вже зазначалося, фаза пріоритезації вимог - це операція, яка визначає пріоритет однієї вимоги порівняно з іншими. Крім того, це найбільш мотивуючий напрямок в області інженерії вимог. На даний момент було запропоновано багато технік для надання переваги вимозі програмного забезпечення порівняно з іншими вимогами у тому ж самому проекті. Техніки пріоритезації вимог класифікуються як порядкова шкала, ординальна шкала і відносна шкала з точки зору потужної пріоритетизації. Найпотужнішою є відносна шкала, яка показує, наскільки одна вимога важливіша за інші вимоги. Найменш потужною є ординальна шкала, яка відповідає за надання вимог в порядку пріоритету та визначає, яка вимога важливіша за інші, але без вказівки на те, наскільки саме важлива.

Аналітичний ієрархічний процес (AHP) є найпопулярнішою і традиційною технікою, яка відноситься до класу відносної шкали. Тому його згадують у великій кількості досліджень [15-17]. Крім того, його вважають системною технікою прийняття рішень, яка використовується для пріоритетизації вимог для конкретного проекту [18-19]. Він порівнює всі можливі пари вимог для їх упорядкування і визначення, яка має вищий пріоритет. AHP не підходить для проектів, у яких є велика кількість вимог [20-21]. Тому багато дослідників намагалися зменшити кількість порівнянь, наприклад, [22-23], і були запропоновані різні техніки для зменшення кількості порівнянь приблизно на 75%, такі як [24].

Кумулятивне голосування (Cumulative Voting) називають "тестом на 100 доларів", що є прямолінійною технікою, де стейкхолдерам надається 100 уявних одиниць, які слід розподілити серед наданих вимог до програмного забезпечення [25]. Результати пріоритетизації подаються у вигляді відносної шкали.

Механізм MoScoW вважається одним з видів числового призначення, який зображений у [26]. Крім того, ця техніка має чотири класи пріоритету: "повинно бути", "може бути", "повинно бути", "не повинно бути". "Повинно бути" означає, що вимоги, які потрапили в цей клас, повинні бути реалізовані спочатку, і так далі. "Може бути" означає, що вимога, яка потрапила в цей клас, існує, і це буде чудово для програмного продукту. "Повинно бути" означає, що вимоги, які знаходяться в цьому класі, повинні бути реалізовані і будуть чудовими для програмного продукту, і, нарешті, "не повинно бути" означає, що вимоги, які потрапили в цей клас, не можуть бути реалізовані на даному етапі, оскільки інші вимоги мають менший пріоритет.

Механізм "сортування бульбашкою" використовується для ранжування елементів, таких як вимоги. У цій техніці вибирають дві вимоги для порівняння між собою. У разі, якщо вимога не впорядкована, їх обмінюють місцями і потім порівнюють з іншою вимогою, і так продовжують, поки не отримають впорядкований список вимог у спадаючому порядку (від вищого до нижчого), як використовується в цих дослідженнях [27].

Техніка бінарного пошукового дерева - це ще один вид методу ранжування, який був запропонований у [27]. Крім того, ця техніка була вперше представлена [15] для пріоритетизації вимог. Кожен вузол у цій техніці вказує на вимогу програмного забезпечення, де всі вимоги, що знаходяться ліворуч від дерева, мають менший пріоритет порівняно з іншими вузлами, тоді як інші вимоги, що знаходяться праворуч від дерева, мають вищий пріоритет. Однак спочатку обирається одна вимога, яка стає кореневим вузлом, та порівнюється з несортованою вимогою. У разі, якщо ця вимога має менший пріоритет, ніж корінь, вона шукає ліворуч від дерева. В іншому випадку вона шукає праворуч від дерева. Операція повторюється, поки не отримаємо відсортоване дерево.

У техніці "десять найважливіших вимог" стейкхолдери вибирають десять найважливіших вимог за їхньою важливістю для них, не вказуючи внутрішнього рангу серед цих вимог. Це робить цю техніку відповідною для багатьох стейкхолдерів однакової важливості [28].

В роботі [11] автор використовував оптимізаційний алгоритм GWO, який є одним з найбільш актуальних метаевристичних алгоритмів, для пріоритетизації вимог програмного проекту. Крім того, цей алгоритм був оцінений і порівняний із механізмом AHP, де запропонована робота показала кращі результати, ніж традиційний метод AHP приблизно на 30%. З іншого боку, в [29] автор застосовував алгоритм WOA, який знедавна використовується для вирішення проблем оптимізації, оскільки він імітує поведінку кита, застосовуючи метод полювання за допомогою бульбашкової сітки. Цей метод також був оцінений за допомогою AHP, де результати показали, що запропонована робота перевершує механізм AHP приблизно на 40%.

Формулювання проблеми: в умовах постійно зростаючої потреби в міждисциплінарних компетенціях існує ряд прикладних задач з прогнозування, зокрема, економічного, серед яких визначення бажаного часу перебування покупців в торговельному залі для отримання найбільшої середньої суми чеку, а також визначення залежності сезонних коливань середньої суми чеку із урахуванням щорічного зростання цін, розв’язання яких можливе засобами машинного навчання.

Мета: розробка програмного коду на основі бібліотек Python для прогнозування зазначених вище явищ. Вибір мови продиктований її поширеністю, доступністю до вільного і безкоштовного використання навіть в комерційних цілях [7], відносною легкістю опанування, можливістю виконувати досить складні вправи з програмування без необхідності встановлення платного або складного середовища.

Задачі:

1. Аналіз тематичних наукових джерел і методичних матеріалів.
2. Експериментальна перевірка запропонованих в них прикладів, їхня адаптація під зазначені вище завдання з прогнозування.
3. Розробка і тестування програмного коду моделей прогнозування.
4. Аналіз результатів, отриманих від розроблених моделей.

МАТЕРІАЛИ ТА МЕТОДИ ДОСЛІДЖЕНЬ

В роботі використано методи структурного і порівняльного аналізу, з інформаційним і аналітичним підходом розглянуто наукову та методичну літературу, а також онлайн ресурси, застосовано експериментальні методи для перевірки програмного коду.

РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Далі розглянуті основні методи, які використані для розв’язання поставленої задачі.

А. ПРІОРИТЕТИЗАЦІЯ ВИМОГ

Значення пріоритетизації вимог розглядається з різних точок зору. Самервіль визначив пріоритетизацію вимог як одне з найважливіших завдань для прийняття рішень [15]. З іншого боку, Файрсміт визнає її основним процесом у сфері програмної інженерії, оскільки вона визначає ідеальний порядок реалізації вимог для планування версій програми та надання бажаної функціональності якнайшвидше. Це також процес визначення пріоритету вимог для зацікавлених сторін на основі їх важливості [16]. Отже, ми можемо зробити висновок, що пріоритетизація вимог передбачає визначення пріоритету за важливістю або за можливістю впровадження.

Реалізація найважливіших операцій, які дозволяють отримати приростовий зворотний зв'язок від клієнта, встановлює графік, виправляє помилки та усуває будь-які непорозуміння між клієнтом і корпорацією на ранніх етапах, призводячи до задоволеності клієнта. Крім того, це корисно для вилучення непотрібних вимог, які можуть бути неефективно витратні, і вибору найбільш підходящих вимог для кожної версії; це сприяє майбутньому плануванню, зменшує ризик скасування, оцінює переваги, розставляє пріоритети в інвестиціях та визначає фінансові наслідки щодо впровадження кожної вимоги [17].

Пріоритетизація вимог є найважливішою і критичною частиною аналізу вимог через обмеження ресурсів проекту. Іншими словами, важко реалізувати всі вимоги одночасно через обмеження ресурсів, таких як графік, персонал і бюджет. Більше того, для удосконалення деяких проектів можуть знадобитися кілька місяців або навіть років, тому важливо визначити вимогу, яка повинна бути реалізована спочатку. Крім того, бюджет відіграє важливу роль, особливо при взаємодії з процесом пріоритетизації вимог, оскільки бюджет вважається менш важливою діяльністю в контексті інженерії вимог порівняно з іншими аспектами програмної інженерії. Наведене вище вказує на те, що вимоги мають різні рівні важливості, і важко визначити, яка з них є найважливішою.

Як було зазначено вище, зацікавлені сторони проекту є основою процесу пріоритетизації з урахуванням бізнес- і регуляторних факторів, оскільки вони мають різні точки зору, і кожен з них повинен визначити найвищий пріоритет вимог, щоб змусити зацікавлені сторони чітко зібрати всі відносно важливі вимоги, що сприяє підвищенню комунікації між ними, надає розумну основу для взаємодії щодо вимог та дозволяє планувати інженерні активності в розумний спосіб.

Б. АЛГОРИТМ GWO

Алгоритм GWO – це один із методів оптимізації, натхненних природою, який був запропонований Мірджалілі та ін. у 2014 році [14]. Він імітує поведінку сірих вовків на полюванні. Основною метою алгоритму GWO є визначення оптимального рішення для задачі за допомогою популяції пошукових агентів.

Ці вовки зазвичай живуть у групах, кількість учасників яких коливається від п'яти до дванадцяти. Основна відмінність між алгоритмом GWO та іншими алгоритмами

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451666

оптимізації полягає в соціальній ієрархії, яка розвиває рішення-кандидата на кожній ітерації оптимізації. На практиці GWO імітує поведінку вовків у пошуку та нападі на жертв на полюванні. Соціальна ієрархія у групі вовків показана на Рис. 1 [14].

Альфа відображає лідера, який є найкращим кандидатом на рішення. Крім того, альфа - домінуючі вовки, за якими йдуть інші. Бета представляє другого кандидата на рішення, який допомагає альфа в прийнятті рішень і є містком між лідером та рештою загону, що грає на найнижчих рівнях. Дельта відображає третього кандидата на рішення, який відповідає за надсилання інформації на два вищих рівні (альфі і беті). Омега представляє решту рішень і відповідає за надсилання інформації на три вищих рівні.

Фактично механізм полювання включає три етапи: відстеження, оточення і напад на здобич. Таким чином, GWO відображає математичний метод полювання сірих вовків, який використовується для вирішення складних задач оптимізації. В даному контексті, оптимальне рішення проблеми розглядається як жертва.

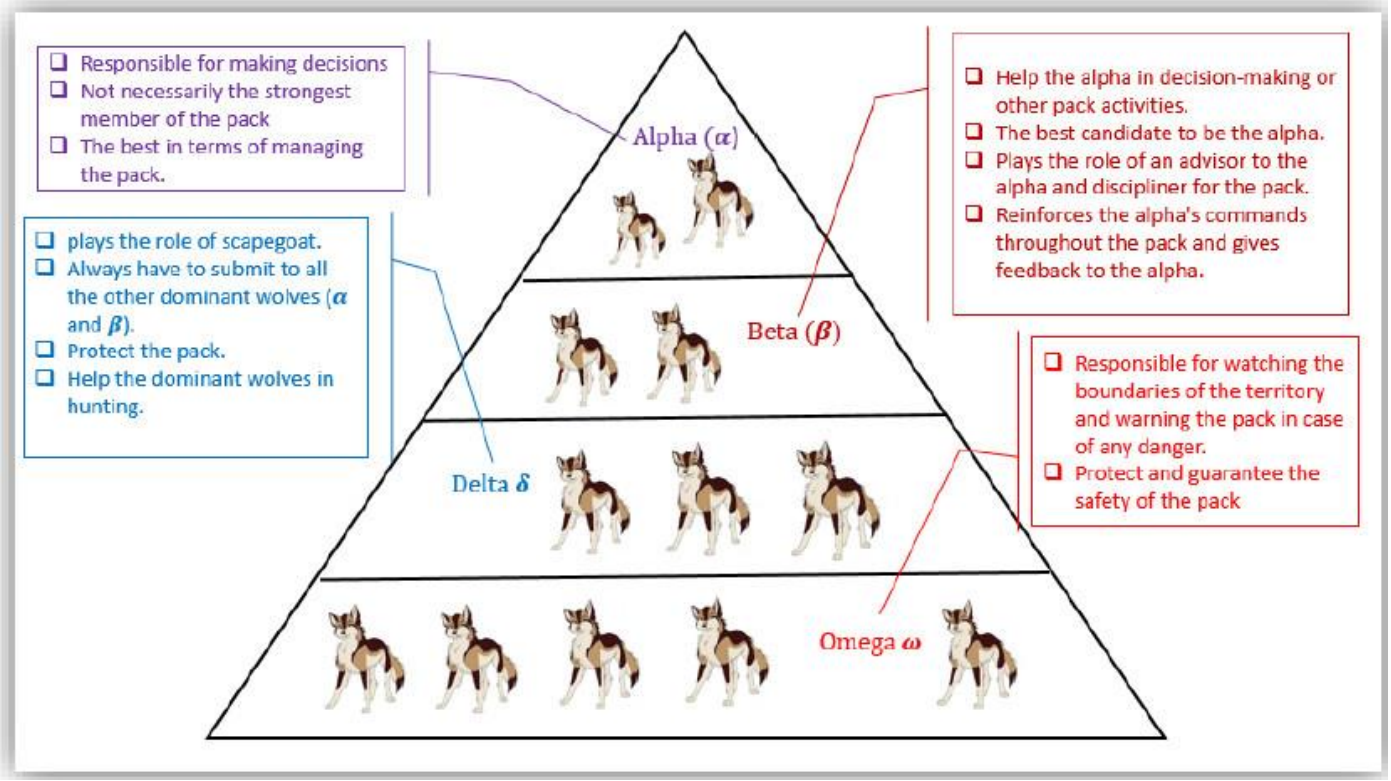


Рис.1. Ієрархія сірих вовків [14]

Рух трьох верхніх рівнів імітує оточення жертви сірими вовками, що відображено у наступній формулі (1):

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_p(t) - \vec{X}(t) \right|, \vec{C} = 2\vec{r}_2 \quad (1)$$

де

t – поточна ітерація,

X_p – вказує на вектор позиції здобичі,

X – позиція сірого вовка,

C – вектор коефіцієнтів,

r_2 – обирається випадково з інтервалу [0, 1].

Таким чином, результат вектора D використовується для зміщення конкретного елементу в бік або від області, в якій знаходиться найкраще рішення, що представляє здобич, за допомогою наступного рівняння (2):

$$\vec{X}(t+1) = \left| \vec{X}_p(t) - \vec{A} \cdot \vec{D} \right|, \vec{A} = 2\vec{a}\vec{r}_1 - \vec{a} \quad , (2)$$

де

r_1 – обирається випадково з інтервалу [0, 1],

a – мінімізується від 2 до 0 протягом попередньо заданої кількості ітерацій.

Щоб побачити ефекти рівнянь (1) та (2), на рис. 2 (а) проілюстровано двовимірний вектор положення та деякі з можливих наступних положень вовка. Як видно на цьому рисунку, сірий вовк у положенні (X, Y) може оновити своє положення відповідно до положення здобичі $(X*, Y*)$. Додавши різні значення навколо найкращого агента до поточної позиції, можна налаштувати значення A та C вектори. Наприклад, $(X* - X, Y*)$ можна отримати, встановивши A = (0, 1) та C = (1, 1).

 https://doi.org/10.5281/zenodo.10451666

Можливі оновлені позиції сірого вовка в тривимірному просторі зображені на Рис. 2 (b). Зверніть увагу, що випадкові вектори і дозволяють вовкам досягти будь-якого положення між точками, зображеними на Рис. 2. Отже, сірий вовк може оновити своє положення всередині простору, та навколо здобичі у будь-якому випадковому місці, використовуючи рівняння (1) та (2).

Цю ж концепцію можна поширити на простір пошуку з n розмірами, і сірі вовки рухатимуться в гіперкубах (або гіперсферах) навколо найкращого рішення, отриманого за зданими умовами [14].

Б.1. ФАЗА ПОЛЮВАННЯ

Сірі вовки мають здатність розпізнавати місцезнаходження здобичі та оточувати їх. Полювання, як правило, керується альфою. Бета та дельта також можуть час від часу брати участь у полюванні. Однак в абстрактному просторі пошуку ми не маємо уявлення про розташування оптимуму (здобичі). Для математичного моделювання мисливської поведінки сірих вовків ми вважаємо, що альфа (найкращий варіант рішення) бета та дельта мають кращі знання про потенційне розташування здобичі. Тому ми зберігаємо перші три найкращі рішення, отримані на сьогодні, та зобов'язуємо інші пошукові агенти (включаючи омеги) оновити свої позиції відповідно до позиції найкращого пошукового агента.

Таким чином, три вищі рівні α, β і δ обчислюються за допомогою наступних математичних виразів (3-5):

$$\vec{D}_\alpha = \left|\vec{C}_1 \cdot \vec{X}_\alpha - X\right|, \vec{X}_1 = \vec{X}_\alpha - \vec{A}_\alpha \cdot (\vec{D}_\alpha), \quad (3)$$

$$\vec{D}_\beta = \left|\vec{C}_2 \cdot \vec{X}_\beta - X\right|, \vec{X}_2 = \vec{X}_\beta - \vec{A}_\beta \cdot (\vec{D}_\beta), \quad (4)$$

$$\vec{D}_\delta = \left|\vec{C}_3 \cdot \vec{X}_\delta - X\right|, \vec{X}_3 = \vec{X}_\delta - \vec{A}_\delta \cdot (\vec{D}_\delta). \quad (5)$$

Для математичного імітування процесу полювання сірого вовка припускається, що α, β і δ мають достатньо інформації про можливе розташування жертви. Більше того, перші три найкращі рішення, які отримані, зберігаються і змушують інших агентів оновлювати свої позиції відповідно до кращих агентів α, β і δ. Ця поведінка математично представлена наступним виразом (6), і псевдокод GWO показаний в Алгоритмі 1 [14].

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3}, \quad (6)$$

Алгоритм 1

Ініціалізація популяції **Xi (i = 1, 2, 3… n)**
Ініціалізація а**, A, C**
Розрахунок придатності для кожного агента
Xα = кращий пошуковий агент
Xβ = **2-**й кращий пошуковий агент
Xδ = **3-**й кращий пошуковий агент
WHILE (t < максимальна к**-**сть ітерацій**)**
  FOR EACH пошукового агента
    оновлення позицій для поточного пошукового агента за рівнянням **(6)**
  ENDFOR
  Оновлення а**, A, C**
  Розрахунок придатності для кожного агента
  Оновлення **X**α**, X**β**, X**δ
  t = t + 1
ENDWHILE
RETURN Xα

На Рис. 3 показано, як пошуковий агент оновлює свою позицію відповідно до альфи, бети та дельти у 2D пошуковому просторі. Можна помітити, що кінцева позиція знаходитиметься у випадковому місці в колі, яке визначається положеннями альфи, бети та дельти в просторі пошуку. Іншими словами, альфа, бета та дельта оцінюють положення здобичі, а інші вовки випадково оновлюють свої позиції навколо здобичі.

Б.2 ФАЗА НАПАДУ

Як згадувалося вище, сірі вовки закінчують полювання, атакуючи здобич, коли вона перестає рухатися. Для математичного моделювання наближення жертви ми зменшуємо значення a. Зверніть увагу, що діапазон коливань A також зменшується на a. Іншими словами A, це випадкове значення в інтервалі$[-a, a]$, де а зменшується з 2 до 0 протягом ітерацій. Коли випадкові значення значень знаходяться в [−1, 1], наступна позиція пошукового агента може знаходитись у будь-якій позиції між його поточною позицією та позицією здобичі.

У випадку, якщо |A| < 1, це відповідає стратегії експлуатації і симулює поведінку нападу на здобич. З іншого боку, якщо |A| > 1, це відповідає стратегії дослідження і імітує віддалення вовків від жертви.

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451666

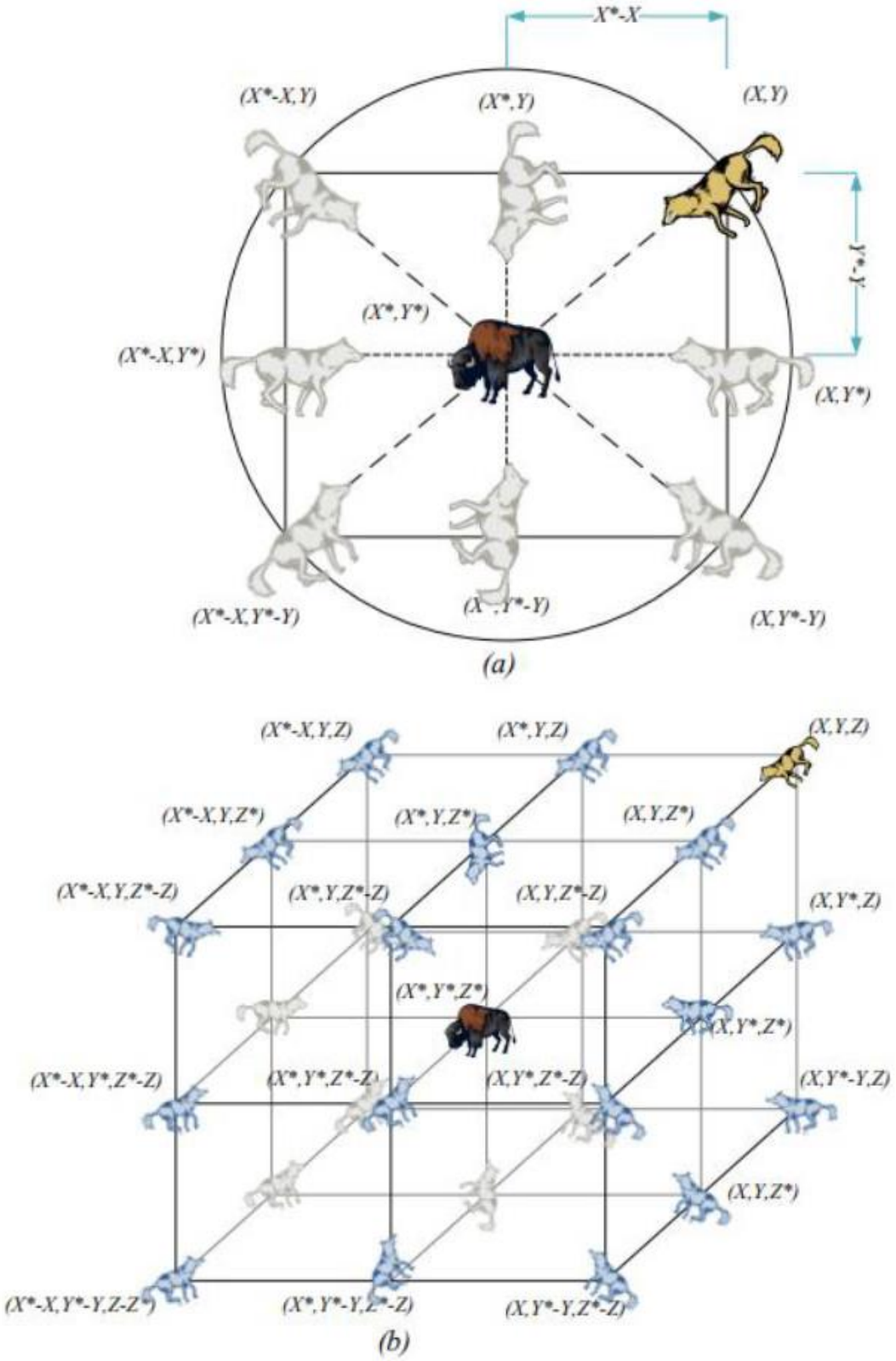


Рис. 2. a) 2D вектор розташування вовка, та одну з можливих наступних локацій [14]
b) 3D вектор розташування вовка, та одну з можливих наступних локацій [14]

 https://doi.org/10.5281/zenodo.10451666

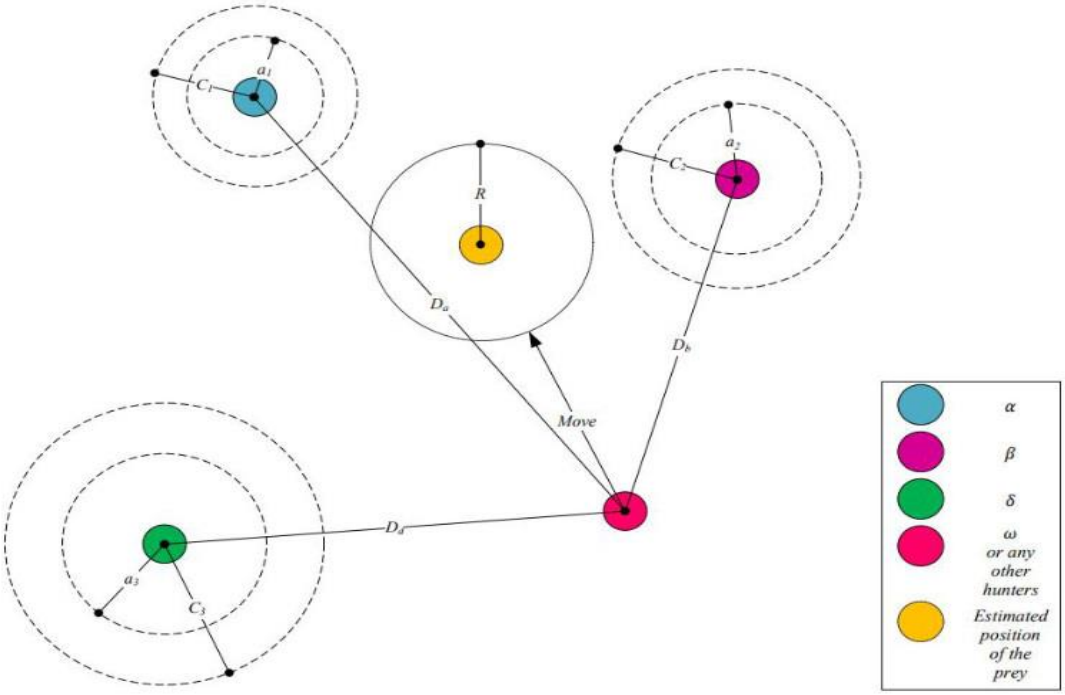


Рис.3. Оновлення позицій в GWO [14]

B. АЛГОРИТМ WOA

Алгоритм WOA є алгоритмом оптимізації, запропонованим Мірджалілі та Льюїсом у 2016 році [10]. Метою цього алгоритму є визначення глобального оптимуму для проблеми за допомогою популяції агентів пошуку (китів). Процес пошуку починається зі створення колекції рішень-кандидатів, які випадковим чином вибираються для задачі. Потім ця колекція поліпшується протягом багатьох ітерацій до досягнення задоволення кінцевої умови. Насправді, кити імітують особливий метод полювання, який називається методом "bubble-net feeding" (метод міхурової сітки), який показаний на Рис. 4 [10].

З Рис. 4 очевидно, що кит зганяє здобич, рухаючись за спіральним маршрутом навколо жертв, утворюючи бульбашки попереду. Таким чином, цей процес пошуку є основним натхненням для WOA. Крім того, ще одним імітованим методом у WOA є метод скорочення оточення.

Рис.4. Поведінка горбатих китів та метод міхурової сітки [10]

Кити-горбачі оточують жертв навколо себе, щоб розпочати полювання на них за допомогою методу ловлі "bubble-net" (метод бульбашкової сітки).

B.1. ФАЗА ОТОЧЕННЯ

Оскільки положення оптимальної схеми в просторі пошуку апріорі невідомо, алгоритм WOA передбачає, що поточне найкраще можливе рішення є цільовою жертвою або близько до оптимального. Після визначення кращого пошукового агента інші пошукові агенти спробують поновити свої позиції щодо кращого пошукового агента. Це поведінка представлено наступними рівняннями [10]:

$$\vec{D} = \left|\vec{C} \cdot \vec{X}^*(t) - \vec{X}(t)\right|, \vec{C} = 2\vec{r}_2 \quad (7)$$

$$\vec{X}(t+1) = \left|\vec{X}^*(t) - \vec{A} \cdot \vec{D}\right|, \vec{A} = 2\vec{a}\vec{r}_1 - \vec{a} \quad (8)$$

де

t – означає поточну ітерацію,

A – коефіцієнтний вектор,

C – коефіцієнтний вектор,

$X*(t)$ – вектор розташування найкращого рішення, отриманого на даний момент,

$X(t)$ – вектор розташування кита,

a – лінійно зменшується від 2 до 0 в процесі ітерацій,

r_1 – випадковий вектор з проміжку [0, 1],

r_2 – випадковий вектор з проміжку [0, 1].

Рис. 5 ілюструє обґрунтування рівняння (7) для двовимірної задачі. Положення (X, Y) пошукового агента

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451666

може бути оновлено відповідно до положення поточної кращого запису ($X*$, $Y*$). Різні місця навколо кращого агента можуть бути досягнуті щодо поточної позиції, регулюючи значення A і C векторів.

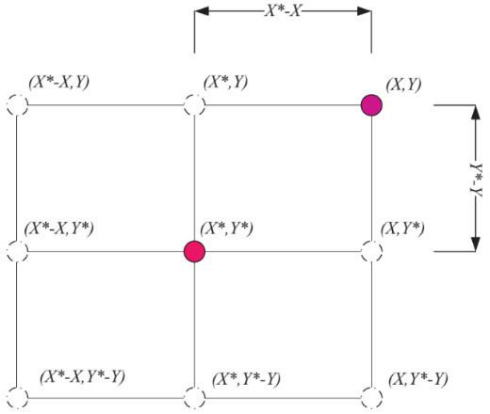


Рис.5. 2D вектори позицій і їх можливі наступні положення (X*, Y* - краще рішення, отримане на даний момент) [10]

Можлива позиція поновлення пошукового агента в тривимірному просторі також зображена на рис. 6.

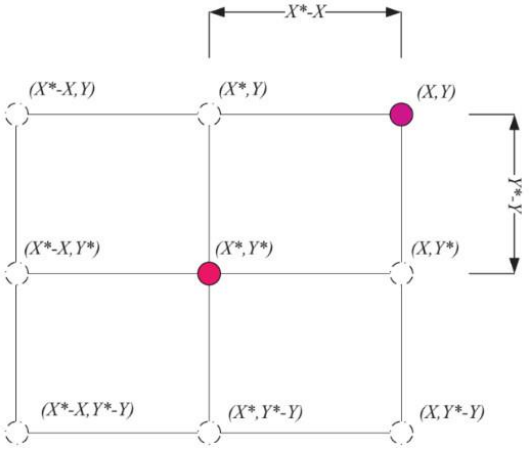


Рис.6. 3D вектори позицій і їх можливі наступні положення (X*, Y*, Z* - краще рішення, отримане на даний момент) [10]

Слід зазначити, що шляхом визначення випадкових векторів r_1 та r_2 можна досягти будь-якої позиції в пошуковому просторі, розташованої між ключовими точками, показаними на Рис. 5 та Рис. 6.

Отже, рівняння (8) дозволяє будь-якому пошуковому агенту оновлювати свою позицію навколо поточного кращого рішення, тим самим – імітує оточення жертви. Ту ж концепцію можна поширити на простір пошуку з n вимірами, і пошукові агенти будуть переміщатися у гіперкубі(ах) навколо кращого рішення, отриманого на даний момент.

Як згадувалося раніше, горбаті кити також атакують здобич, використовуючи стратегію пухирчастої сітки. Для математичного моделювання поведінки пухирчатої сітки горбатих китів розроблено два підходи: механізми скорочення оточення та пошуку за допомогою бульбашкової сітки.

Механізм скорочення оточення: така поведінка досягається зменшенням значення a в рівнянні (8). Зверніть увагу, що діапазон коливань A також зменшується на a. Іншими словами, A є випадковим значенням в інтервалі [$-a$, a], де a зменшується з 2 до 0 протягом ітерацій. Встановлюючи випадкові значення для A в діапазоні [-1,1], нову позицію пошукового агента можна визначити де завгодно, між початковою позицією агента та позицією поточного найкращого агента.

На Рис. 7 показано можливі положення від (X, Y) до (X∗, Y∗), яких можна досягти за допомогою 0≤ A ≤1 у двовимірному просторі.

Механізм спірального оновлення позицій. Як видно на Рис. 8, цей підхід спочатку обчислює відстань між китом, що знаходиться в (X, Y), і здобиччю, що знаходиться в (X∗, Y∗). Потім створюється спіральне рівняння між положенням кита і здобиччю, щоб імітувати спіральний рух горбатих китів, як показано нижче:

$$\vec{X}(t+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}*(t) \quad (9),$$

де

b – постійна, що визначає форму логарифмічної спіралі,

l – вказує на випадкове число в [−1, 1],

D' = |X∗ (t) – X (t)| - вказує на відстань між і-м китом та жертвою.

Зверніть увагу, що горбаті кити плавають навколо здобичі по звужуючому колі і по спіральній траєкторії одночасно. Щоб змоделювати цю одночасну поведінку, ми припускаємо, що існує 50%-а ймовірність вибору між механізмом скорочення оточення або спіральної моделі для оновлення положення китів в процесі оптимізації.

Це поведінка математично представлена наступними рівняннями (10):

$$\vec{X}(t+1) = \begin{cases} \vec{X}*(t) - \vec{A} \cdot \vec{D}, \text{якщо } p < 0,5 \\ D' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}*(t), \text{якщо } p \geq 0,5 \end{cases} \quad (10)$$

де

p – це випадкове число в [0, 1],

t – представляє поточну ітерацію.

 https://doi.org/10.5281/zenodo.10451666

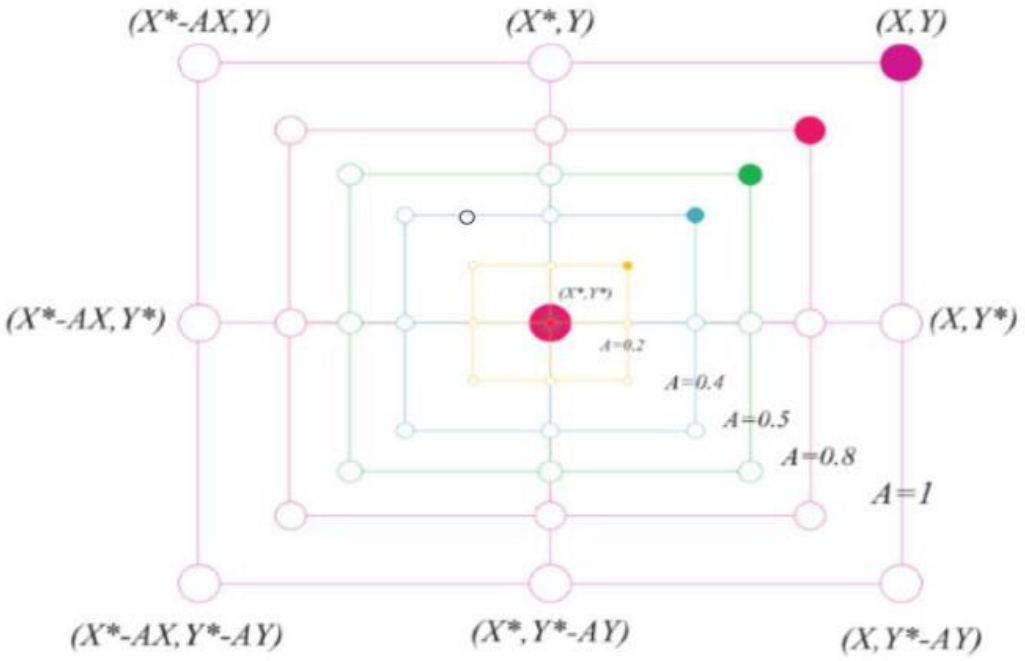


Рис.7. Механізм пошуку за допомогою бульбашкової сітки, реалізований в WOA (X* - найкраще рішення, отримане на даний момент), механізм скорочення оточення [10]

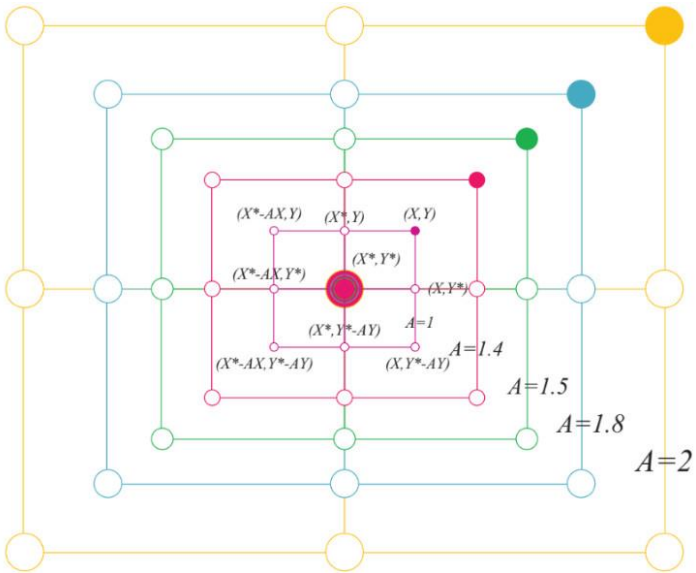


Рис.8. Механізм дослідження, реалізований в WOA (X* - це випадково обраний пошуковий агент)

B.2. ФАЗА РОЗВІДКИ

У разі, якщо задано проблему, WOA починає оптимізацію цієї проблеми, генеруючи колекцію випадкових рішень. На кожній ітерації агенти оновлюють своє розташування в залежності від випадково вибраного агента або найкращого агента, який виграв до цього часу. Для забезпечення фази дослідження інші агенти оновлюють свої позиції на основі найкращого рішення, яке опорною точкою, коли |A| > 1. У іншому випадку, коли |A| < 1, найкраще рішення виконує іншу роль з опорною точкою (Рис. 8).

Математична модель виглядає наступним чином (11-12):

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451666

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_{rand} - \vec{X}(t) \right| , (11)$$

$$\vec{X}(t+1) = \left| \vec{X}_{rand} - \vec{A} \cdot \vec{D} \right| , (12)$$

де

t – означає поточну ітерацію,

A – коефіцієнтний вектор,

$Xrand$ – вектор випадково вибраного кита (із поточної популяції).

Псевдокод WOA показано в Алгоритмі 2 [10].

Алгоритм 2

Ініціалізація китової популяції X_i $(i = 1, 2, \ldots n)$
Розрахунок придатності кожного агента
X^* = найкращий пошуковий агент
WHILE (*t* < максимальна к-сть ітерацій**)**
FOR EACH пошукового агенту
ОНОВЛЕННЯ ***a***, ***A***, ***C***, ***l*** та ***p***
IF1 (*p* < 0.5)
IF2 (|*A*| < 1)
Оновлення вектора позицій поточного пошукового агента за рівнянням **(7)**
ELSE IF2 (|*A*| ≥ 1)
Випадково вибрати пошукового агента **(xrand)**
Оновлення вектора позицій поточного пошукового агента за рівнянням **(12)**
ENDIF2
ELSE IF1 (*p* ≥ 0.5)
Оновлення вектора позицій поточного пошукового агента за рівнянням **(9)**
ENDIF1
ENDFOR
Перевірка кожного пошукового агента на вихід за границі
Розрахунок придатності кожного агента
Оновлення X*
t = t + 1
ENDWHILE
RETURN X^*

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

ЗАПРОПОНОВАНИЙ АЛГОРИТМ (HYGWWOA)

Підхід, запропонований у цьому дослідженні, в основному розроблений шляхом поєднання алгоритмів WOA та GWO з метою створення гібридного підходу, який називається "HyGWWOA". Процес поєднання здійснюється шляхом об'єднання переваг кожного з алгоритмів, які використовуються в цьому дослідженні. Основною перевагою GWO є те, що з кожного кластеру вибираються три найвищих значення, які вважаються α, β і δ, і ці значення представляють три найкращі рішення з цього кластеру, з урахуванням обмеження кількості учасників у групі в середньому від 5 до 12 вузлів. З іншого боку, у WOA немає обмежень для кількості учасників у групі, і, нарешті, у WOA немає топ-три найкращих рішень, а замість цього - одне. Враховуючи вищенаведене, HyGWWOA пропонується здійснити поєднання цих двох алгоритмів, застосовуючи алгоритм WOA для уникнення обмежень кількості учасників у групі та обираючи топ-три найвищих рішення, як це робить GWO.

Як і будь-який метаевристичний оптимізаційний алгоритм, запропонований алгоритм складається з двох етапів. Перший - це етап дослідження, тоді як другий - етап експлуатації.

У цьому дослідженні запропонований алгоритм HyGWWOA використовується для пошуку в усьому просторі пошуку з метою знаходження всіх можливих Xrand. Ці Xrand відіграють важливу роль в етапі дослідження. З іншого боку, якщо будь-який агент пошуку бачить здобич, вважається, що це Xrand, і цей процес пошуку здійснюється за допомогою створення бульбашок по шляху, який утворює форму спіралі, і його називають технікою полювання на мережу бульбашок. Таким чином, коли будь-який агент пошуку бачить бульбашки, це означає, що вони належать до цієї спіралі. У випадку, якщо одночасно є багато спіралей від різних Xrand, кожна з них розглядається як кластер в просторі пошуку, і в кожному з них є перші три найкращі рішення Xα, Xβ і Xδ. Ця перевага взята з алгоритму GWO, тоді як попередній сценарій представляє етап експлуатації.

На кожній ітерації, після визначення найкращих рішень Xα, Xβ і Xδ, інші агенти пошуку оновлюють свої позиції в кластері. Враховуючи значення абсолютного A, якщо воно більше одиниці, це означає, що агент пошуку виходить за межі кластера (спіралі), і йому потрібно шукати інший Xrand, щоб належати до нового кластера. В іншому випадку абсолютне значення A менше одиниці, що означає, що агент пошуку залишається в кластері і наближається до здобичі.

У цій роботі пропонується покращений підхід, який поєднує нещодавно запропоновані біоінспіровані оптимізаційні техніки, які були розроблені Мірджалілі та іншими, а саме алгоритми GWO і WOA, що імітують техніки полювання в природі. Потім для визначення пріоритетів вимог використовується алгоритм HyGWWOA. Для отримання пріоритету вимог застосовується алгоритм HyGWWOA. Алгоритм 3 представляє рекомендований псевдокод алгоритму HyGWWOA.

Алгоритм 3

BEGIN
Ініціалізувати популяцію агентів X_i $(i = 1, 2, 3 \ldots\ldots n)$

 https://doi.org/10.5281/zenodo.10451666

Ініціалізувати ***C***, ***r*** і ***a***
Випадковим чином вибрати X_{rand}
Розрахувати відстань між кожним китом **(*i*)** та усіма X_{rand} за рівнянням **(1)**
IF агент **(i)** не призначений
призначити агента **(i)** його найближчому X_{rand}
CALL FITNESS для кожного агента **(i)**
CALL CLUSTER
CALL RP
повернути найкраще рішення
ENDIF
END

A. ЕТАП ІНІЦІАЛІЗАЦІЇ

На початку ініціалізуйте популяцію агентів, як показано в Алгоритмі 3. Кількість агентів вибирається випадковим чином для генерації бульбашки, яка виглядає як конус (спіральна форма). Потім виміряйте відстань між усіма агентами та всіма Xrand, щоб призначити їх найближчому і з'єднати цю групу, щоб вона стала її членом.

B. ФІТНЕС-ФУНКЦІЯ

На основі рівняння (13), Xrand створює бульбашкову сітку, коли він бачить жертву, всі агенти, які бачать бульбашкову сітку, будуть асоціюватися з кластером (спіраллю). Отже, інші агенти, які приєдналися до кластера, повинні оновити свої розташування в напрямку розташування Xrand, як показано в Алгоритмі 4. З іншого боку, ці агенти оновлюють свої розташування в залежності від значення абсолютного A. У випадку, якщо |A| менше за одиницю, це означає, що агент все ще входить до кластера і оновлює своє розташування за допомогою рівняння (13). В іншому випадку, якщо |A| більше або дорівнює одиниці, це означає, що агент не належить до кластера, тому він шукає інший Xrand для асоціації, використовуючи рівняння (14). Ця поведінка представлена математично рівняннями (13, 14) та рівнянням (15).

$$\vec{X}(t+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}(t), \quad (13)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D}, \quad (14)$$

$$V = \frac{1}{3}\pi r h, \quad (15)$$

де

r – радіус бульбашкової сітки, який є постійною величиною (r = 12),

h – висота бульбашкової сітки, яка вибирається випадковим чином між 5 і 15.

Алгоритм 4. Функція FITNESS

PROCEDURE FITNESS
BEGIN
X_{rand} створює бульбашкову сітку за допомогою формули **(13)**
FOR EACH агента пошуку **(*i*)**
Оновити ***a***, ***A***, ***C*** та ***L***
Розрахувати відстань між кожним агентом **(*i*)** та X_{rand} за допомогою формули **(1)**
IF (A<1)
Оновити положення поточного агента **(*i*)** за допомогою формули **(13)**
ELSE
Вибрати новий X_{rand} випадковим чином
Оновити положення поточного агента **(*i*)** за допомогою формули **(14)**
ENDIF
ENDFOR
END

C. КЛАСТЕРИЗАЦІЯ

Як показано в Алгоритмі 5, кожний кластер K має Xrand, який вибирається випадковим чином. Функція пристосованості розраховується для кожного агента з метою перевірки, чи бачить цей агент бульбашку, яку створив Xrand. Іншими словами, цей агент все ще приєднаний до цієї спіралі. У випадку, якщо агент не бачить бульбашку, він шукає інший Xrand, до якого можна приєднатися. Таким чином, кожен кластер K отримує Xα, Xβ та Xδ, які представляють перші три найважливіші вимоги у цьому кластері.

Алгоритм 5. Функція CLUSTER

PROCEDURE CLUSTER
BEGIN
FOR EACH кластера *K*
Вибрати X_{rand} випадковим чином
Вибрати X_α, X_β та X_δ
WHILE (*t* ≤ *max_iteration*)
CALL FITNESS
CALL RP
ENDWHILE
RETURN X_α, X_β та X_δ
ENDFOR
END

D. ФУНКЦІЯ ПРІОРИТЕЗАЦІЇ ВИМОГ

Як вже обговорено на етапі кластеризації, кожний агент у просторі пошуку представляє вимогу з її важливістю; ця важливість відображає її значення в процесі розробки цільового проекту критичної інформаційної інфраструктури. Ця важливість розраховується відповідно до ранжування зацікавлених сторін для кожної вимоги з урахуванням важливості цієї зацікавленої сторони та її впливу на проект.

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10451666

Оскільки кожна зацікавлена сторона належить до ролі в цільовому середовищі, для якого будується система, ранг ролі має прямий вплив на важливість зацікавленої сторони. Таким чином, при розрахунках важливості зацікавленої сторони ранг ролі відіграє ключову роль.

Згідно з проектним середовищем, спершу буде розраховано важливість ролі на основі ранжування зацікавлених сторін за рівнем важливості зацікавленої сторони (16):

$$R_{\inf}(i) = \frac{RR_{\max} + 1 - Rank(R(i))}{\sum_{j=1}^{n} RR_{\max} + 1 - Rank(R(j))}, \quad (16)$$

де

$RR_{\max}$ - максимальний ранг в списку ролей,

$Rank(R(i)$ - ранг і-ї ролі,

n – кількість стейкхолдерів з однаковою роллю.

Далі важливість кожної зацікавленої сторони в кожній ролі розраховується для визначення впливу цієї зацікавленої сторони на проект (17):

$$ST_{\inf}(i) = \frac{RS_{\max} + 1 - Rank(i)}{\sum_{K=1}^{n} RS_{\max} + 1 - Rank(K)}, \quad (17)$$

де

і – представляє конкретну зацікавлену сторону,

RS_{max} – максимальний ранг зацікавлених сторін в даній ролі,

Rank(i) - ранг і-ої зацікавленої сторони у ролі,

n – загальна кількість зацікавлених сторін в тій же ролі.

Потім вплив цієї зацікавленої сторони на проект взагалі розраховується шляхом множення впливу ролі та впливу зацікавленої сторони, як показано в (18):

$$PR_{\inf}(i) = R_{\inf}(i) \cdot ST_{\inf}(i), \quad (18)$$

Оскільки кожна зацікавлена сторона ранжує список вимог, важливість цієї вимоги розраховується шляхом сумування впливу всіх зацікавлених сторін на проект, помноженого на її оцінку цієї вимоги (19):

$$R_{imp}(i) = \sum_{i=1}^{n} PR_{\inf}(i) \cdot r(i), \quad (19)$$

де

r(i) – представляє ранг і-ої зацікавленої сторони на цій вимозі,

n – загальна кількість зацікавлених сторін, які ранжують вимогу r(i).

Як показано в Алгоритмі 6, з кожного кластера будуть вибрані три найкращі значення. Цей процес буде виконуватися для кожного кластера ітеративно. Результатом цього процесу буде набір кращих рішень з кожного кластера, який буде збережений у тимчасовому списку. Отже, цей список містить α, β та δ з кластерів. Оскільки цей список містить найкращі рішення, три найкращі рішення будуть вибрані як перший результат процесу пріоритетизації. Вибрані результати будуть переміщені в остаточний відсортований список як три найкращі рішення. Оскільки вони вже вибрані та відсортовані за пріоритетом, ці вимоги будуть видалені зі створених кластерів.

Цей процес буде повторюватися до тих пір, поки в усіх кластерах не залишиться вимог. Як зазначалося вище, вибрані найкращі рішення будуть видалені зі створених кластерів, отже, кількість вимог у кластерах буде зменшуватися з кожною ітерацією.

Кінцевим результатом роботи запропонованого підходу буде повернення остаточного відсортованого набору вимог відповідно до їх важливості.

Алгоритм 6. Функція RP

PROCEDURE PR
BEGIN
WHILE кожен кластер не порожній **(*1,2,3,...*K)**
FOR EACH кластера ***K***
Відсортувати вимоги на основі рівняння **(19)**
Вибрати $\mathbf{X}_{\alpha}$, $\mathbf{X}_{\beta}$ і $\mathbf{X}_{\delta}$ та перемістити їх до **temp_list**
ENDFOR
Вибрати X_{α}, X_{β} і X_{δ} з відсортованого ***temp_list*** та перемістити їх до кінцевого відсортованого списку
Видалити X_{α}, X_{β} і X_{δ} з оригінальних кластерів
ENDWHILE
Повернути кінцевий набір результатів
END

ЕКСПЕРИМЕНТАЛЬНІ РЕЗУЛЬТАТИ

Для оцінки запропонованого методу пріоритетизації вимог щодо точності було вибрано проект SKUDA як об'єкт вивчення в цій статті. SKUDA – це проект оновлення системи доступу до об'єктів критичної інфраструктури (ОКІ).

Багато будівель SKUDA потребують авторизованого доступу, таких як підстанції, серверні та комп'ютерні кластери.

Мета SKUDA полягає в заміні застарілих систем контролю доступу, об'єднанні різних існуючих засобів контролю доступу.

SKUDA – це досить великий проект. В ньому більше 20 груп зацікавлених осіб і приблизно 7 000 користувачів. До деяких з цих груп належали службовці з безпеки, розробники, менеджери та співробітники служб підтримки, таких як Департамент обслуговування та

 https://doi.org/10.5281/zenodo.10451666

управління майном, який відповідає за фізичний стан інфраструктури, Департамент людських ресурсів, що відповідає за інформацію про персонал, Департамент інформаційних технологій. У SKUDA близько 7 000 працівників та відвідувачів, які використовують систему для входу в будівлі та доступу на інфраструктурні об’єкти та отримання доступу до IT.

SKUDA має складну базу зацікавлених осіб, різні з яких мають протилежні вимоги.

У цьому дослідженні реалізовано роботу щодо вимог з урахуванням ранжування цілей проекту та відкинута робота щодо конкретних вимог, оскільки запропонований метод стосується пріоритетизації вимог.

Тестовий набір даних містить 53 вимоги, кожна з яких має свою важливість. Результати роботи показані в Таблиці 1, яка відображає впорядкований список вимог.

Запропонований підхід дає можливість впорядкувати вищевказаний набір вимог за важливістю з точністю 92%.

Таблиця 1. Пріоретизований список вимог

ID	Опис вимоги	Пріоритет
a.3	Все в 1 картці	1
a.1	Легший в використанні	2
a.2	Той самий спосіб контролю доступу для входу до ОКІ	3
c.3	Візуальний контроль	4
c.4	Контроль доступу, включаючи відстеження руху/журнали	5
c.5	Посилення контролю доступу до будівель	6
c.2	Контроль доступу до будівель ОКІ	7
c.1	Забезпечення відповідного доступу для кожної особи	8
d.5	Надання прав доступу	9
d.1	Швидкий випуск карток	10
d.2	Скорочення часу стояння в черзі	11
d.3	Статус ID-картки: можливість перевірити, чи отримав користувач ID-картку	12
d.6	Можливість створювати звіти про доступ	13
d.4	Легка заміна втрачених карток	14
d.7	Процедури боротьби з шахрайством	15
b.3	Картка має бути безпечною	16
b.1	Картка для включення даних користувача	17
b.4	Картка з брендом ОКІ	18
b.5	Легка ідентифікація/картка має чіткий вигляд	19
b.6	Карта має бути міцною/надійною	20
b.2	Картка з QR-кодом	21
b.7	Картка повинна бути привабливою	22
g.1	Централізоване управління інформацією доступу та ідентифікації	23
g.2	Експорт даних в інші системи	24
g.3	Імпорт даних з інших систем	25
g.4	Доступ до даних: можливість переглядати, оновлювати, видаляти віддалено та безпечно	26
g.5	Чіткі правила використання даних доступу	27
e.1	Економія на картках	28
e.2	Економія часу обробки	29
e.3	Скорочення паперових випробувань	30
f.5	Сумісність із поточною мережевою інфраструктурою	31
f.6	Вплив на інші системи	32
f.4	Сумісність з UPI	33
f.2	Сумісність з системами ОКІ	34
f.1	Сумісність з системою SKUDA	35
f.3	Сумісність з HR системою	36
h.3	Використання для входу в комп'ютер	37
h.2	Включення механізму оплати	38
h.4	Можливість оновлення (версії програмного забезпечення)	39
h.1	Додавання цифрового сертифікату	40

ISSN 2786-9024

h.5	Підвищення безпеки	41
j.2	Відповідність стандартам і законодавству	42
j.6	Наявність	43
j.5	Надійність	44
j.3	Технології	45
j.1	Безвідмовність	46
j.7	Мережева інфраструктура	47
j.4	Життєвий цикл	48
j.9	Вибраний виробник повинен мати перевірену історію відстеження в установах із технологіями контролю доступу, виготовлення ідентифікаційних карток, ODBC, смарт-карт.	49
j.8	Здатність прямого друку на обох сторонах картки, яка включатиме QR-код OKI	50
i.3	Діяльність з управління проектами	51
i.2	Технічна документація	52
i.1	Підтримка постачальника	53

ВИСНОВКИ

Інженерія вимог є найважливішою фазою у розробці проектів критичної інфраструктури, оскільки вона має справу з зацікавленими сторонами та іншими активностями. Оскільки кількість вимог змінюється для кожного проекту, процес пріоритизації вимог є важливим для встановлення порядку фаз проекту та задоволення зацікавлених сторін та кінцевих користувачів. У цій роботі було запропоновано гібридний підхід, який полягає в поєднанні переваг алгоритмів GWO та WOA як метаевристичних підходів для пріоритизації вимог проекту критичної інфраструктури.

Ця робота відкриває шлях для подальших досліджень в галузі пріоритизації вимог у сфері розробки проектів критичної інфраструктури та використання метаевристичних підходів для оптимізації процесів вирішення проблем у IT-галузі.

ЛІТЕРАТУРА

[1] M. S. Hasan, A.A. Mahmood, M.J.Alam, S.N.Hasan, & F.Rahman, “An evaluation of software requirement prioritization techniques”, *International Journal of Computer Science and Information Security* (IJCSIS), vol. 8, iss. 9, 2010.

[2] C.Duan, P.Laurent, J.Cleland-Huang, & C.Kwiatkowski, “Towards automated requirements prioritization and triage”, *Requirements engineering*, vol. 14, iss. 2, 73-89, 2009, doi: 10.1007/s00766-009-0079-7.

[3] R.Masadeh, A.Alzaqebah, & A.Sharieh, “Whale Optimization Algorithm For Solving The Maximum Flow Problem”, *Journal of Theoretical & Applied Information Technology*, vol. 96, iss. 8, 2018.

[4] A.K.Jain, M.N.Murty, & P.J.Flynn, “Data clustering: a review”, *ACM computing surveys* (CSUR), vol. 31, iss. 3, 264-323, 2019, doi: 10.1145/331499.331504.

[5] H.Emami, & F.Derakhshan, “Integrating fuzzy K-means, particle swarm optimization, and imperialist competitive algorithm for data clustering”, *Arabian Journal for Science and Engineering*, vol. 40, iss. 12, 3545-3554, 2015, doi: 10.1007/s13369-015-1826-3.

[6] F.Yang, T.Sun, & C.Zhang, “An efficient hybrid data clustering method based on K-harmonic means and Particle Swarm Optimization”, *Expert Systems with Applications*, vol. 36, iss. 6, 9847-9852, 2009, doi: 10.1016/j.eswa.2009.02.003

[7] J.Nayak, B.Naik, & H.S.Behera, “Fuzzy C-means (FCM) clustering algorithm: a decade review from 2000 to 2014”, *Computational intelligence in data mining*, vol. 2, 133-149,2015, doi: 10.1007/978-81-322-2208-8_14. Springer, New Delhi.

[8] Y.Kumar, & G.Sahoo, “Hybridization of magnetic charge system search and particle swarm optimization for efficient data clustering using neighborhood search strategy”, *Soft Computing*, vol. 19, iss. 12, 3621-3645, 2015, doi: 10.1007/s00500-015-1719-0.

[9] Q.H.Zhang, B.L.Li, Y.J.Liu, L.Gao, L.J.Liu, & X.L.Shi, “Data clustering using multivariant optimization algorithm”, *International Journal of Machine Learning and Cybernetics*, vol. 7, iss. 5, 773-782, 2016, doi: 10.1007/s13042-014-0294-5.

[10] S.Mirjalili, & A.Lewis, “The whale optimization algorithm”, *Advances in Engineering Software*, vol. 95, 51-67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[11] A.Alzaqebah, R.Masadeh, & A.Hudaib, “Whale Optimization Algorithm for Requirements Prioritization”, *the 9th International Conference on Information and Communication Systems* (ICICS), 84-89, 2019, doi: 10.1109/IACS.2018.8355446.

[12] S.Chander, P.Vijaya, & P.Dhyani, “Multi kernel and dynamic fractional lion optimization algorithm for data clustering”, *Alexandria engineering journal*, vol. 57, iss. 1, 267-276, 2018, doi: 10.1016/j.aej.2016.12.013.

[13] E.Çomak, “A modified particle swarm optimization algorithm using Renyi entropy-based clustering”, *Neural Computing and Applications*, vol. 27, iss. 5, 1381-1390, 2016, doi: 10.1007/s00521-015-1941-9.

[14] S.Mirjalili, S.M.Mirjalili, & A.Lewis, "Grey wolf optimizer", *Advances in engineering software*, vol. 69, 46-61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.
[15] J.Karlsson, C.Wohlin, & B.Regnell, "An evaluation of methods for prioritizing software requirements", *Information and software technology*, vol. 39, iss. 14-15, 939-947, 1998, doi: 10.1016/S0950-5849(97)00053-0.
[16] D.Leffingwell, D.Widrig, "Managing Software Requirements: A Unified Approach". Upper Saddle River:Addison- Wesley, 2009.
[17] S.Hatton, "Early prioritisation of goals", *International Conference on Conceptual Modeling*, 235-244, 2007, doi: 10.1007/978-3-540-76292-8_29. Springer, Berlin, Heidelberg.
[18] B.Regnell, M.Höst, J.Natt och Dag, P.Beremark, T.Hjelm, "An industrial case study on distributed prioritization in market-driven requirements engineering for packaged software", *Requirements Engineering*, vol. 6, iss. 1, 51-62, 2001, doi: 10.1007/s007660170015.
[19] T.L.Saaty, "The analytic hierarchy process". McGraw-Hill, New York, 1980.
[20] L.Lehtola, & M.Kauppinen, "Empirical evaluation of two requirements prioritization methods in product development projects", *European Conference on Software Process Improvement*, 161-170, doi: 10.1007/978-3-540-30181-3_15. Springer, Berlin, Heidelberg, 2004.
[21] L.Lehtola, "Suitability of requirements prioritization methods for market-driven software product development", *Softw. Process Improve. Pract.*, vol. 11, 7-19, 2006, doi: 10.1002/spip.249.
[22] Y.Shen, A.E.Hoerl, & W.Mcconnell, "An incomplete design in the analytic hierarchy process", *Mathematical and Computer Modelling: An International Journal*, vol. 16, iss. 5, 121-129, 1992.
[23] P.T.Harker, "Incomplete pairwise comparisons in the analytic hierarchy process" , *Mathematical Modelling*, vol. 9, iss. 11, 837-848, 1998, doi: 10.1016/0270-0255(87)90503-3.
[24] J.Karlsson, S.Olsson, & K.Ryan, "Improved practical support for large-scale requirements prioritising", *Requirements Engineering*, vol. 2, iss. 1, 51-60, 1997, doi: 10.1007/BF02802897.
[25] D.Leffingwell & D.Widrig, "Managing Software Requirements: A Unified Approach". Upper Saddle River:Addison- Wesley, 1999.
[26] DSDM Project Framework. [Online]. URL: http://surl.li/rlcbf. Accessed: 09.11.2023.
[27] A.V.Aho, J.E.Hopcroft, & J.Ullman, "Data structures and algorithms". Addison-Wesley Longman Publishing Co., Inc., 1993.
[28] S.Lauesen, "Software requirements – styles and techniques". Pearson Education, Essex, 2002.
[29] R.Masadeh, A.Alzaqebah & A.Hudaib, "Grey Wolf Algorithm for Requirements Prioritization", *Modern Applied Science*, vol. 12, iss. 2, 54, 2018, doi: 10.5539/mas.v12n2p.
[30] S.L.Lim, "Social networks and collaborative filtering for large-scale requirements elicitation", doctoral dissertation, University of New South Wales, 2011. [Online]. URL: http://surl.li/rlcck. Accessed: 09.11.2023.

REQUIREMENT PRIORITIZATION IN THE DEVELOPMENT OF SOFTWARE PROJECTS FOR CRITICAL INFRASTRUCTURE OBJECTS

Iaroslav Dorohyi, Olena Doroha-Ivaniuk

The objective of the study is to develop an algorithm for prioritizing requirements in the development of software for critical infrastructure object projects. Requirement development is a fundamental phase in any software project, as this phase involves the identification, processing, and manipulation of requirements. The primary source of these requirements is project stakeholders, taking into account project constraints and limits. The number of requirements varies for each software project for a critical infrastructure object, hence the term requirement prioritization pertains to determining the priority order of executing software requirements based on considerations and decisions of stakeholders.

Various proposed optimization algorithms are employed to address optimization tasks. This paper presents the main stages of basic optimization algorithms, some of their modifications aimed at enhancing their efficiency in solving such types of problems. Additionally, a hybrid approach based on WOA and GWO optimization algorithms is proposed, combining the advantages of each algorithm to determine the priority of requirements for critical infrastructure object software. Furthermore, a dataset from the SKUDA project is provided, utilized in this research, meeting the requirements of a real software project for evaluating the proposed method.

The scientific novelty lies in the modification, application, and combination of results from well-known GWO and WOA algorithms to address the requirement prioritization task for critical infrastructure object software projects. The proposed algorithm achieves an accuracy of 92% for the proposed set of requirements.

Keywords: *requirement prioritization, WOA (Whale Optimization Algorithm), GWO (Grey Wolf Optimization), critical infrastructure object, CI (Critical Infrastructure), hybrid approach.*

REFERENCES

[1] M. S. Hasan, A.A. Mahmood, M.J.Alam, S.N.Hasan, & F.Rahman, "An evaluation of software requirement prioritization techniques", *International Journal of Computer Science and Information Security* (IJCSIS), vol. 8, iss. 9, 2010.
[2] C.Duan, P.Laurent, J.Cleland-Huang, & C.Kwiatkowski, "Towards automated requirements prioritization and triage", *Requirements engineering*, vol. 14, iss. 2, 73-89, 2009, doi: 10.1007/s00766-009-0079-7.
[3] R.Masadeh, A.Alzaqebah, & A.Sharieh, "Whale Optimization Algorithm For Solving The Maximum Flow Problem", *Journal of Theoretical & Applied Information Technology*, vol. 96, iss. 8, 2018.

ISSN 2786-9024

[4] A.K.Jain, M.N.Murty, & P.J.Flynn, "Data clustering: a review", *ACM computing surveys* (CSUR), vol. 31, iss. 3, 264-323, 2019, doi: 10.1145/331499.331504.

[5] H.Emami, & F.Derakhshan, "Integrating fuzzy K-means, particle swarm optimization, and imperialist competitive algorithm for data clustering", *Arabian Journal for Science and Engineering*, vol. 40, iss. 12, 3545-3554, 2015, doi: 10.1007/s13369-015-1826-3.

[6] F.Yang, T.Sun, & C.Zhang, "An efficient hybrid data clustering method based on K-harmonic means and Particle Swarm Optimization", *Expert Systems with Applications*, vol. 36, iss. 6, 9847-9852, 2009, doi: 10.1016/j.eswa.2009.02.003

[7] J.Nayak, B.Naik, & H.S.Behera, "Fuzzy C-means (FCM) clustering algorithm: a decade review from 2000 to 2014", *Computational intelligence in data mining*, vol. 2, 133-149,2015, doi: 10.1007/978-81-322-2208-8_14. Springer, New Delhi.

[8] Y.Kumar, & G.Sahoo, "Hybridization of magnetic charge system search and particle swarm optimization for efficient data clustering using neighborhood search strategy", *Soft Computing*, vol. 19, iss. 12, 3621-3645, 2015, doi: 10.1007/s00500-015-1719-0.

[9] Q.H.Zhang, B.L.Li, Y.J.Liu, L.Gao, L.J.Liu, & X.L.Shi, "Data clustering using multivariant optimization algorithm", *International Journal of Machine Learning and Cybernetics*, vol. 7, iss. 5, 773-782, 2016, doi: 10.1007/s13042-014-0294-5.

[10] S.Mirjalili, & A.Lewis, "The whale optimization algorithm", *Advances in Engineering Software*, vol. 95, 51-67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[11] A.Alzaqebah, R.Masadeh, & A.Hudaib, "Whale Optimization Algorithm for Requirements Prioritization", *the 9th International Conference on Information and Communication Systems* (ICICS), 84-89, 2019, doi: 10.1109/IACS.2018.8355446.

[12] S.Chander, P.Vijaya, & P.Dhyani, "Multi kernel and dynamic fractional lion optimization algorithm for data clustering", *Alexandria engineering journal*, vol. 57, iss. 1, 267-276, 2018, doi: 10.1016/j.aej.2016.12.013.

[13] E.Çomak, "A modified particle swarm optimization algorithm using Renyi entropy-based clustering", *Neural Computing and Applications*, vol. 27, iss. 5, 1381-1390, 2016, doi: 10.1007/s00521-015-1941-9.

[14] S.Mirjalili, S.M.Mirjalili, & A.Lewis, "Grey wolf optimizer", *Advances in engineering software*, vol. 69, 46-61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[15] J.Karlsson, C.Wohlin, & B.Regnell, "An evaluation of methods for prioritizing software requirements", *Information and software technology*, vol. 39, iss. 14-15, 939-947, 1998, doi: 10.1016/S0950-5849(97)00053-0.

[16] D.Leffingwell, D.Widrig, "Managing Software Requirements: A Unified Approach". Upper Saddle River:Addison- Wesley, 2009.

[17] S.Hatton, "Early prioritisation of goals", *International Conference on Conceptual Modeling*, 235-244, 2007, doi: 10.1007/978-3-540-76292-8_29. Springer, Berlin, Heidelberg.

[18] B.Regnell, M.Höst, J.Natt och Dag, P.Beremark, T.Hjelm, "An industrial case study on distributed prioritization in market-driven requirements engineering for packaged software", *Requirements Engineering*, vol. 6, iss. 1, 51-62, 2001, doi: 10.1007/s007660170015.

[19] T.L.Saaty, "The analytic hierarchy process". McGraw-Hill, New York, 1980.

[20] L.Lehtola, & M.Kauppinen, "Empirical evaluation of two requirements prioritization methods in product development projects", *European Conference on Software Process Improvement*, 161-170, doi: 10.1007/978-3-540-30181-3_15. Springer, Berlin, Heidelberg, 2004.

[21] L.Lehtola, "Suitability of requirements prioritization methods for market-driven software product development", *Softw. Process Improve. Pract.*, vol. 11, 7-19, 2006, doi: 10.1002/spip.249.

[22] Y.Shen, A.E.Hoerl, & W.Mcconnell, "An incomplete design in the analytic hierarchy process", *Mathematical and Computer Modelling: An International Journal*, vol. 16, iss. 5, 121-129, 1992.

[23] P.T.Harker, "Incomplete pairwise comparisons in the analytic hierarchy process" , *Mathematical Modelling*, vol. 9, iss. 11, 837-848, 1998, doi: 10.1016/0270-0255(87)90503-3.

[24] J.Karlsson, S.Olsson, & K.Ryan, "Improved practical support for large-scale requirements prioritising", *Requirements Engineering*, vol. 2, iss. 1, 51-60, 1997, doi: 10.1007/BF02802897.

[25] D.Leffingwell & D.Widrig, "Managing Software Requirements: A Unified Approach". Upper Saddle River:Addison- Wesley, 1999.

[26] DSDM Project Framework. [Online]. URL: http://surl.li/rlcbf. Accessed: 09.11.2023.

[27] A.V.Aho, J.E.Hopcroft, & J.Ullman, "Data structures and algorithms". Addison-Wesley Longman Publishing Co., Inc., 1993.

[28] S.Lauesen, "Software requirements – styles and techniques". Pearson Education, Essex, 2002.

[29] R.Masadeh, A.Alzaqebah & A.Hudaib, "Grey Wolf Algorithm for Requirements Prioritization", *Modern Applied Science*, vol. 12, iss. 2, 54, 2018, doi: 10.5539/mas.v12n2p.

[30] S.L.Lim, "Social networks and collaborative filtering for large-scale requirements elicitation", doctoral dissertation, University of New South Wales, 2011. [Online]. URL: http://surl.li/rlcck. Accessed: 09.11.2023.

 https://doi.org/10.5281/zenodo.10473684

Інформаційна безпека

КОНЦЕПТУАЛЬНІ ЗАСАДИ ГЛОБАЛЬНОЇ СТІЙКОСТІ СМАРТ-ДЕРЖАВИ

Я.Ю. Дорогий[1], І.О. Бердиченко[2]

[1] Department of Applied Mathematics and Informatics, Donetsk National Technical University, Lutsk, Ukraine
[2] Department of Criminal Law and Procedure, Kyiv University of Law of the National Academy of Sciences of Ukraine
E-mail: yaroslav.dorohyi@donntu.edu.ua

Отримано 31.12.2023
Прийнято до публікації 19.01.2024
Опубліковано 01.04.2024

АНОТАЦІЯ

Ця стаття присвячена розгляду концептуальних засад глобальної стійкості смарт-держави. Автори розглядають сучасний вимір розвитку України у контексті впровадження технологій та інновацій у всі сфери життя. Зокрема, стаття розглядає вплив інформаційних технологій, штучного інтелекту, та інших сучасних технологій на ефективність управління, соціально-економічний розвиток та екологічну сталість.

У статті детально розглядаються основні аспекти створення та функціонування смарт-держав, включаючи роль цифрових інфраструктур, відкритих даних, та електронного уряду. Автори аналізують принципи взаємодії смарт-держави з громадянами, підприємствами та іншими учасниками суспільства для досягнення високого рівня стійкості та розвитку.

Стаття також розглядає виклики та ризики, пов'язані з впровадженням сучасних технологій у державну систему, та пропонує стратегії забезпечення кібербезпеки та захисту приватності в умовах цифрового середовища. Особлива увага приділяється ідеям сталого розвитку та етичним аспектам використання технологій для досягнення глобальної стійкості смарт-держав, розумінню таких дефініцій як цифрова стійкість та глобальна стійкість, і відповідно, визначенню їх структури і закономірностей застосування.

***Ключові слова**: глобальна стійкість, смарт-держава, стійкість, цифрова інфраструктура, сталість, концептуальні засади*

ВСТУП

У сучасному світі стрімкого технологічного прогресу та глибоких змін у соціально-економічній сфері виникає необхідність перегляду та удосконалення парадигм управління державами. Однією з ключових концепцій, що визначає сучасний розвиток, є ідея створення смарт-держав, яка базується на впровадженні інформаційних технологій, штучного інтелекту та інновацій для досягнення вищого рівня ефективності, сталості та розвитку.

Концепція смарт-держави виникла в кінці 1990-х років і стала активно розвиватися в останні роки. Це пов'язано з рядом факторів, включаючи:

- швидкий розвиток технологій, таких як Інтернет, штучний інтелект та блокчейн;
- зростаюче значення сталого розвитку;
- зміна ролі держави в суспільстві.

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10473684

Смарт-держава – це держава, яка використовує технології для підвищення ефективності управління, соціально-економічного розвитку та екологічної сталості. Вона характеризується такими основними рисами:

- цифрова інфраструктура, яка забезпечує доступ до інформації та послуг для громадян, підприємств та інших учасників суспільства;
- відкриті дані, які дозволяють використовувати інформацію для прийняття рішень та вирішення проблем;
- електронний уряд, який забезпечує доступ до державних послуг в онлайн-режимі.

Впровадження смарт-держави пов'язане з низкою викликів та ризиків, таких як:

- кібербезпека: загрози порушення конфіденційності та цілісності даних;
- захист приватності: загрози збору та використання персональних даних без згоди громадян;
- нерівність: загрози посилення нерівності в суспільстві через доступ до технологій.

Для забезпечення кібербезпеки та захисту приватності необхідно розробляти чіткі нормативні вимоги, інвестувати у розвиток кібербезпеки та захищати приватність громадян. Для вирішення проблеми нерівності необхідно забезпечити доступ до технологій для всіх громадян, незалежно від їхнього соціального статусу та фінансового становища. До того ж, подолання зазначених викликів вимагає дотримання принципу верховенства права і принципу людиноцентричності в діяльності суб’єктів владних повноважень.

Смарт-держави мають потенціал для створення більш стійких і справедливих суспільств. Технології можуть бути використані для вирішення таких глобальних проблем, як зміна клімату, бідність та голод.

АНАЛІЗ ЛІТЕРАТУРНИХ ДАНИХ ТА ПОСТАНОВКА ПРОБЛЕМИ

Розглянемо деякі літературні джерела та нормативно-правові акти України, які визначають основну термінологію та місце смарт-держави у розвитку України.

У дослідженні [1] розглядається роль цифрової трансформації у формуванні смарт-держави на прикладі європейського досвіду. Автори визначають, що цифрові технології сприяють перетворенню відносин між державою і громадянами, зокрема підвищенню ефективності управління та соціально-економічного розвитку, і відповідно, створенню успішної інноваційної держави. Вони стверджують, що цифрові трансформації є одним із ключових факторів успішної реалізації цього завдання.

Стаття [2] розглядає проблему кібербезпеки як важливого аспекту глобальної стійкості смарт-держави. Автори визначають, що кібербезпека – це стан захисту інформації та систем від несанкціонованого доступу, використання, розкриття, модифікації або знищення. Вони стверджують, що кібербезпека є ключовим фактором забезпечення ефективності управління, соціально-економічного розвитку та екологічної сталості смарт-держави.

В науковій праці [3] розглядаються етичні аспекти використання технологій у формуванні смарт-держави. Автори визначають, що етичні аспекти використання технологій - це сукупність моральних норм і принципів, які регулюють діяльність людей у сфері технологій. Вони стверджують, що етичні аспекти є важливим фактором забезпечення сталого розвитку смарт-держави.

У статті [4] надається огляд літератури з питань смарт-управління для сталого розвитку. Автори визначають, що смарт-управління - це використання технологій для підвищення ефективності управління та досягнення цілей сталого розвитку. Вони стверджують, що смарт-управління має потенціал для вирішення таких глобальних проблем, як зміна клімату, бідність та голод.

У статті [5] надається огляд літератури з питань смарт-управління для сталого розвитку міст. Автори визначають, що смарт-управління містами - це використання технологій для підвищення ефективності управління містами та досягнення цілей сталого розвитку. Вони стверджують, що смарт-управління містами має потенціал для вирішення таких проблем, як забруднення навколишнього середовища, транспортна інфраструктура та доступність житла.

1 серпня 2022 року European Commission оприлюднила результати Індексу цифрової економіки та суспільства за 2022 рік (Digital Economy and Society Index 2022 (DESI)). Індекс цифрової економіки та суспільства (DESI) – це зведений індекс, який узагальнює відповідні показники з ефективності цифрових технологій у Європі та відстежує еволюцію держав-членів ЄС в області цифрової конкурентоспроможності. Він зараз складається з таких компонентів [6]:

- людський капітал (human capital);
- зв’язок (connectivity);
- інтеграція цифрових технологій (integration of digital technology);
- цифрові державні послуги (digital public setvices).

Далі наведено огляд джерел, які є важливими документами, що визначають напрямок розвитку України в найближчі роки. Вони відображають прихильність

українського уряду до інновацій, цифрової трансформації та сталого розвитку.

У контексті теми, що розглядається, ці джерела мають важливе значення. Вони демонструють, що Україна прагне стати сучасною державою, яка використовує технології для підвищення ефективності управління, соціально-економічного розвитку та екологічної сталості:

- програма діяльності Кабінету Міністрів України [7] передбачає підвищення ефективності державного управління, в тому числі за рахунок використання технологій;
- національна стратегія із створення безбар’єрного простору в Україні [8] сприятиме підвищенню доступності технологій для всіх громадян, незалежно від їхнього соціального статусу та фінансового становища;
- державна стратегія регіонального розвитку [9] передбачає використання технологій для розвитку регіонів України;
- стратегія реформування державного управління [10] передбачає підвищення прозорості та підзвітності державного управління, що можна досягти за рахунок використання технологій;
- план дій із впровадження Ініціативи «Партнерство «Відкритий Уряд» [11] сприятиме відкритості та прозорості діяльності органів державної влади;
- розпорядження Кабінет Міністрів України від 17 лютого 2021 р. № 365-р «Деякі питання цифрової трансформації» [12] передбачає створення єдиної цифрової платформи для взаємодії органів державної влади, бізнесу та громадян;
- концепція розвитку системи електронних послуг в Україні [13] передбачає розширення доступу до державних послуг в онлайн-режимі;
- концепція розвитку штучного інтелекту в Україні [14] передбачає використання штучного інтелекту для вирішення таких проблем, як зміна клімату, бідність та голод;
- стратегія розвитку сфери інноваційної діяльності на період до 2030 року [15] передбачає створення сприятливого середовища для розвитку інновацій, включаючи технології;
- дорожня карта реформування ІТ-освіти [16] передбачає підготовку кваліфікованих кадрів для ІТ-індустрії;
- розпорядження КМУ від 12 травня 2021 р. № 438-р «Про затвердження плану заходів з реалізації Концепції розвитку штучного інтелекту в Україні на 2021-2024 роки» [17] передбачає створення сприятливого середовища для розвитку штучного інтелекту в Україні;
- стратегія кібербезпеки України [18] передбачає захист даних та інфраструктури від кіберзагроз;
- концепція забезпечення національної системи стійкості [19] передбачає використання технологій для підвищення стійкості України до зовнішніх загроз.

У цілому, наведені джерела демонструють, що Україна має амбітні плани щодо розвитку смарт-держави. Реалізація цих планів потребуватиме значних зусиль та ресурсів, але вона має потенціал для створення більш ефективної, справедливої та стійкої держави.

Отож приходимо до висновку, що цифрові трансформації, що охопили світ мають за мету посилення спроможностей щодо забезпечення цифрової та кібербезпеки держави, її цифрового простору, підтримки цифровими засобами та технологіями соціальної та політичної стабільності, оборони держави, захисту державного суверенітету та територіальної цілісності, конституційного ладу, забезпечення прав та свобод кожного громадянина.

Мета статті полягає в дослідженні та розкритті концептуальних засад глобальної стійкості смарт-держави. Ця стаття ставить за мету систематизацію та аналіз ключових аспектів створення та функціонування смарт-держав, зосереджуючись на їхньому впливі на ефективність управління, соціально-економічний розвиток, та екологічну сталість. Додатково, стаття пропонує вивчення викликів і ризиків, пов'язаних із застосуванням інформаційних технологій у державному управлінні, та розглядає стратегії забезпечення кібербезпеки та захисту приватності в умовах цифрового середовища. Крім того, стаття висвітлює ідеї сталого розвитку та етичні аспекти використання технологій для досягнення глобальної стійкості смарт-держав.

Результатом роботи є висновки та рекомендації щодо оптимального впровадження концепції смарт-держав у сучасному світі.

МАТЕРІАЛИ ТА МЕТОДИ ДОСЛІДЖЕНЬ

В роботі використано методи структурного і порівняльного аналізу, з інформаційним і аналітичним підходом розглянуто наукову та методичну літературу, а також онлайн ресурси.

РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Реалізація таких трансформацій потребує комплексного підходу, і отже на наш погляд, може бути реалізовано через розробку і впровадження в практичну площину Концепції глобальної стійкості Смарт-Держави (далі — Концепція), яка б визначила мету, основні принципи, напрями, механізми і строки запровадження

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10473684

та функціонування системи глобальної стійкості, спрямованої на забезпечення здатності держави, бізнесу і суспільства своєчасно ідентифікувати загрози, виявляти вразливості та оцінювати ризики, запобігати або мінімізувати їх негативні впливи, ефективно реагувати та швидко повномасштабно відновлюватися після виникнення загроз або настання надзвичайних та кризових ситуацій усіх видів, включаючи загрози гібридного типу, але не обмежуючись ними, постійно адаптуватися та трансформуватися шляхом впровадження цифрових технологій в усі процеси та на всіх рівнях діяльності держави та суспільства.

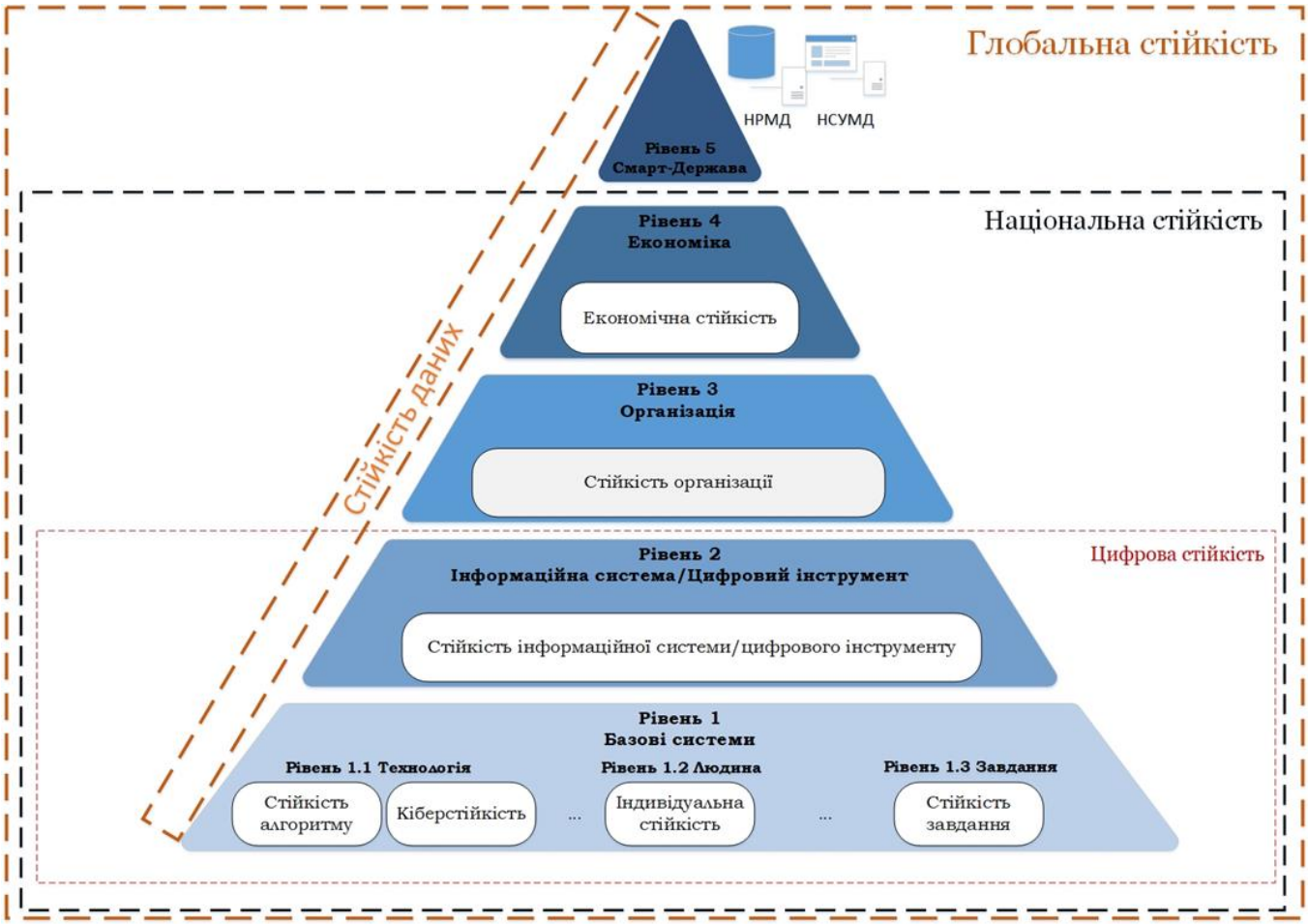


Рис. 1. Концепція глобальної стійкості

Концепція глобальної стійкості Смарт-Держави спрямована на забезпечення здатності держави, бізнесу і суспільства шляхом впровадження цифрових технологій та інструментів на всіх рівнях швидко відновлюватися до первинного стабільного стану або до іншого стабільного стану без втрат або з отриманням зиску і передбачає запровадження відповідних цифрових інструментів на рівні: держави (національна економіка), організації, інформаційної системи/цифрового інструменту та базових систем. Структура рівнів представлена на Рис. 1.

Рівень 1 та 2 передбачають впровадження цифрових інструментів та технологій для досягнення цифрової стійкості. Забезпечення стійкості базових систем рівня 1 напрямỳ впливає на стійкість інформаційних систем та цифрових інструментів рівня 2. Для прикладу, обізнаність людини щодо питань, пов’язаних з кібергігієною, безпосередньо впливає на стійкість її персонального кабінету в банківській установі; стек використовуваних технологій та заходи з кібербезпеки безумовно впливають на стійкість інформаційної системи, яка на них побудована, і таке інше.

Цифрову стійкість можна охарактеризувати за допомогою наступних чотирьох ключових компонентів: кібербезпеки, безперервності процесів, захисту персональних даних та цифрового громадянства.

1) Кібербезпека складається зі стандартів, практики та людських ресурсів, необхідних для підтримки функціонування цифрових систем та забезпечення стану захищеності цифрової екосистеми. Вона включає систему управління ризиками, яка дозволяє особам, що

приймають рішення, розраховувати величину ризику, пов’язаного з цифровими системами, і регулярно підтримувати можливості, достатні для прогнозування та реагування на інциденти та надзвичайні ситуації на постійній основі.

2) Безперервність процесів передбачає планування та можливості для управління кризовими ситуаціями та відновлення, які практикуються для забезпечення того, щоб державні установи та бізнес-організації продовжували функціонувати навіть у несприятливих умовах. Безперервність залежить від наявності відповідних правил та стандартів, які дають бізнесу можливість для проведення фінансових операцій, забезпечуючи при цьому швидку адаптацію в рамках передбачуваного та загальноприйнятого набору правил та передової практики.

3) Захист персональних даних включає надійну екосистему даних, що складається з законів, установ і можливостей, які визначають і регулюють збір, зберігання та видалення даних. Функціонально це базується на визначенні прав доступу і використання, і тому, як дані, включаючи персональні дані, збираються і використовуються державою, підприємствами та іншими третіми сторонами. Приватність та захист даних важливі для запобігання збиткам, забезпечення цілісності державних та ділових операцій та захисту цифрового громадянства від потенційних зловживань, несправедливих рішень або помилок, а також для забезпечення економічної діяльності.

4) Цифрове громадянство означає готовність і здатність громадян безпечно користуватися перевагами цифрових систем та інфраструктури. Цифрове громадянство включає базову цифрову грамотність, практичне застосування цифрової гігієни та навичок, що забезпечують особисту безпеку в цифровому середовищі, а також поінформованість про цифрові права та обов'язки при використанні цифрових систем та даних.

Цілеспрямоване впровадження та використання вказаних компонентів цифрової стійкості на рівні організації шляхом впровадження «правильних» цифрових інструментів дозволяє досягти вже стійкості рівня організації (рівень 3). На цьому рівні також передбачається вжиття заходів, спрямованих на посилення відмовостійкості та запобігання загрозам технологічної залежності від іноземних виробників і постачальників продукції, технологій та послуг, що забезпечують функціонування цифрової екосистеми організації/підприємства/установи.

Стійке підприємство/організація/установа є фундаментальним елементом побудови стійкої економіки (рівень 4). На цьому рівні вживаються комплекси цілеспрямованих дій, методів та механізмів взаємодії «правильних» органів державної влади, органів місцевого самоврядування, підприємств, установ, організацій, інститутів громадянського суспільства, які гарантують збереження безпеки і безперервності функціонування основних сфер життєдіяльності суспільства і держави до, під час і після настання кризової ситуації. Фактично досягається стан національної стійкості.

Останній крок – трансформація стійкої держави у Смарт-Державу. Для забезпечення такого роду трансформації потрібно досягти стану стабільності даних, який передбачає впровадження на всіх рівнях та у всіх сферах діяльності держави національної системи управління майстер-даними (НСУМД) та національного репозиторію майстер-даних (НРМД).

Майстер-дані – вичерпний перелік сутностей організації/підприємства/установи та їх атрибутів, необхідний для надання інформації та цифрових послуг громадянам. Наприклад, до майстер даних можуть відноситися: реєстрова інформація, програмне забезпечення, класифікатори, стандарти, нормативна база тощо.

НСУМД являє собою сукупність процесів та інструментів для постійного визначення та централізованого управління основними даними держави (майстер-даними) та забезпечить:

- створення єдиного інформаційного простору;
- зв'язування інформації про одні і ті ж же майстер-дані з різних систем;
- збільшення ефективності бізнес-процесів та ділових процесів, пов'язаних з майстер-даними;
- зменшення часу обробки майстер-даних та мінімізацію державних витрат;
- оптимізацію зусиль щодо відповідності бізнесу регуляторним вимогам;
- швидке повернення інвестицій у розрізі роботи з майстер-даними;
- зниження загальної вартості володіння.

Впровадження НРМД дозволить державі постійно мати у використанні «правильне» представлення даних з інформаційних джерел завдяки:

- дослідженню даних – аналіз, профілювання та оцінка майстер-даних у інформаційних джерелах;
- стандартизації даних – приведення майстер-даних до єдиного формату;
- зіставленню даних – підрахунок схожості, виявлення дублів, кандидатів для злиття;
- видаленню дублікатів даних – створення "золотого" запису та збагачення;

ISSN 2786-9024

– інтеграції даних – online/offline інтеграція з існуючими бізнес-застосунками, платформою «Дія» та іншими державними застосунками;

– безпеці даних – централізований контроль та безпека майстер-даних.

Інтеграція НСУМД та НРМД у повсякденну діяльність держави дозволить досягти стану глобальної стійкості – фінального стану цифрової трансформації держави, при якому досягається її здатність забезпечувати належний рівень надання послуг громадянам в будь-який момент часу та в будь-якій точці незалежно від типу, виду та масштабу загроз, які на неї впливають, а самій державі легко влитися у міжнародний інформаційний простір.

Побудована за таким принципом держава і є Смарт-Державою, яка характеризується станом глобальної стійкості і здатна виходити з будь-якої кризи з зиском для себе.

Проблема, яка потребує розв’язання, обмумовлена тим, що глобальні тенденції розвитку цифрової економіки в умовах мінливого ландшафту загроз вимагають переосмислення традиційних підходів до безпеки. Кібербезпека та захист інформації самі по собі не можуть забезпечити сталість цифрового розвитку. Вони стали частиною більш широкої концепції «цифрова стійкість», яка орієнтована на запобігання та адаптивність, та включає питання управління ризиками цифрового розвитку.

Ризик-інформований підхід, покладений в основу цифрової стійкості, забезпечує систематичне виявлення потенційних вразливостей інформаційних ресурсів, систем, мереж та загроз для них, імовірнісного оцінювання виникнення негативних подій, детерміністичного оцінювання потенційних негативних наслідків цих подій та розроблення рекомендацій щодо реалізації контрзаходів з метою мінімізації вразливостей, імовірностей виникнення негативних подій та їхніх наслідків.

Викладене дозволяє нам запропонувати наступне визначення цифрова стійкість - підхід, спрямований на забезпечення здатності держави, бізнесу і суспільства своєчасно ідентифікувати загрози, виявляти вразливості та оцінювати ризики, запобігати або мінімізувати їх негативні впливи, ефективно реагувати та швидко і повномасштабно відновлюватися після виникнення загроз або настання надзвичайних та кризових ситуацій усіх видів, включаючи загрози гібридного типу, але не обмежуючись ними, постійно адаптуватися та трансформуватися шляхом впровадження цифрових технологій в усі процеси та на всіх рівнях діяльності держави та суспільства.

Цифрова стійкість є елементом більш комплексного підходу «глобальної стійкості», який вимагає активної участі всіх зацікавлених сторін, включаючи державу, бізнес та громадянське суспільство.

Фактично, глобальна стійкість – це набір можливостей, методів та сприятливих умов, які забезпечують безперервність діяльності держави, бізнесу та суспільства в умовах постійних змін. Виникає необхідність концептуалізувати глобальну стійкість як набір стратегій, практик, можливостей та інструментів, які допомагають передбачати, запобігати та реагувати на неминучі економічні і геополітичні кризи, природні та техногенні катастрофи, і як результат - виходити з них з зиском для держави.

Запровадження на основі національних інтересів України та з урахуванням міжнародного досвіду багаторівневої комплексної системи глобальної стійкості сприятиме формуванню на державному, регіональному та місцевому рівнях необхідних спроможностей для запобігання та належного реагування держави, бізнесу і суспільства на широкий спектр загроз та швидкого відновлення після кризових ситуацій шляхом постійного адаптування та трансформування ділових та бізнес-процесів через впровадження нових цифрових технологій та інструментів в усі процеси діяльності та на всіх рівнях «Смарт-Держави».

Комплексне бачення мети, процедур, детальний перелік завдань та заходів, якісних і кількісних показників, очікуваних результатів реалізації завдань з розбудови цифрової стійкості на прикладі держави Україна, пропонуємо визначити через розроблення відповідної Концепції глобальної стійкості Смарт-держави. Метою реалізації Концепції є визначення основних принципів, напрямів, механізмів і строків запровадження та функціонування системи цифрової стійкості, спрямованої на забезпечення здатності держави, бізнесу і суспільства своєчасно ідентифікувати загрози, виявляти вразливості та оцінювати ризики, запобігати або мінімізувати їх негативні впливи, ефективно реагувати та швидко і повномасштабно відновлюватися після виникнення загроз або настання надзвичайних та кризових ситуацій усіх видів, включаючи загрози гібридного типу, але не обмежуючись ними, постійно адаптуватися та трансформуватися шляхом впровадження цифрових технологій в усі процеси та на всіх рівнях діяльності держави та суспільства.

Іншими словами, метою держави є створення сприятливого (у тому числі нормативно-правового) середовища та можливостей цифрового розвитку для всіх сторін (самої держави, бізнесу, громадянського суспільства) з повною інтеграцією у світовий простір.

 https://doi.org/10.5281/zenodo.10473684

Результатом має стати власне досягнення стану глобальної стійкості в усіх сферах життя, а сама держава має трансформуватися у «Смарт-Державу». Концепцію, що передбачає такі масштабні перетворення в державі, доцільно розробляти на період біля 10 років.

Реалізацію завдань з побудови цифрової стійкості будь якої державного утворення, на наш погляд, доцільно реалізувати через нижченаведені організаційні та практичні заходи, отже наведемо нашу позицію на прикладі побудови цифрової стійкості держави Україна.

Для досягнення вказаної мети на державному рівні необхідно:

– визначити принципи та політику цифрової стійкості та безпеки,

– здійснити аудит чинного законодавства та законодавчих бар’єрів, закласти основу створення комплексного нормативно-правового підходу до зазначеної сфери переглянути, і відповідно модернізувати законодавство, розробити відповідні національні стандарти;

– гармонізувати чинне законодавство відповідно до міжнародних стандартів

– розробити дорожню карту реалізації проекту «Цифрова стійкість України», орієнтовно на період від 3 до 5 років;

– розробити та впровадити дорожні карти розвитку цифрової стійкості для найбільш актуальних галузей економіки та сфер життєдіяльності (освіта, медицина, транспорт, енергетика тощо)

– впровадити алгоритм «Цифровий фільтр» у повсякденну діяльність, який передбачає обов’язкової процедури аналізу на наявність цифрових варіантів реалізації тих чи інших проектів при прийнятті будь-яких рішень, ініціатив національного, регіонального, галузевого рівнів та їх відповідності критеріям та показникам цифрової стійкості;

– створити та впровадити національну системи управління майстер-даними (НСУМД), національний репозиторій майстер-даних (НРМД) та національний резервний центр обробки даних (НРЦОД);

– наділити функціями формування політики центральний орган виконавчої влади, що відповідає за державну політику у сфері цифрових трансформацій, та який стане відповідальним за глобальну стійкість;

– забезпечити стійкість національних мереж та цифрових активів, у тому числі критичної інформаційної інфраструктури;

– розробити та впровадити нові програми підготовки та навчання фахівців, які працюють у критичних секторах економіки та державного управління;

– стимулювати державні програми щодо впровадження доступу до широкосмугового Інтернету по всій державі;

– створити, модернізувати та надати доступ до національних хмарних ресурсів для бізнесу;

– продовжити активне переведення державних послуг в онлайн формат (насамперед важливих для бізнесу – оподаткування, ліцензування та реєстрація тощо);

– підтримати цифрову освіту, забезпечити надійний доступ до централізованих онлайнових освітніх ресурсів для вчителів та учнів, забезпечити доступ до обладнання та відповідного програмного забезпечення, а також до високоякісного широкосмугового зв'язку;

– на системній основі проводити інформування та навчання (усіх) у галузі цифрової гігієни;

– діджіталізувати та розширити доступ до медичних послуг;

– розробити державні програми та прискорити доступ до цифрових транзакцій (фінансово-технічної, «пісочниці», електронна торгівля тощо).

Результатом реалізації Концепції повинна стати побудова сучасної та ефективної системи цифрової стійкості для забезпечення підтримки і подальшого цифрового розвитку ефективної та прозорої системи управління державою в цілому.

ВИСНОВКИ

Цифрова стійкість є ключовим фактором забезпечення безпеки та стабільності держави в умовах сучасного світу. Запропонована Концепція глобальної стійкості Смарт-Держави є всебічною та продуманою програмою, яка спрямована на підвищення рівня цифрової стійкості держави у всіх сферах її діяльності. Реалізація Концепції потребуватиме значних ресурсів та зусиль, але вона дозволить побудувати стійку та успішну державу в цифрову епоху.

ЛІТЕРАТУРА

[1] J.Dupont. The Smart State. Redesigning government in the era of intelligent services. London: Policy Exchange 8 – 10 Great George Street, 2018. 45 p. [Онлайн]. URL: http://surl.li/rlcqa. Дата звернення: 06.01.2024.

[2] І.В.Васильєва, О.Ю.Губенко, “Кібербезпека як складова глобальної стійкості смарт-держави”, *Наукові праці Національного університету «Львівська політехніка». Серія «Інформаційні технології та засоби обчислювальної техніки»*, 293, 121-132, 2022.

[3] Ю.О.Губенко, І.В.Васильєва, “Етичні аспекти використання технологій у формуванні смарт-держави”, *Наукові праці Національного університету «Львівська*

ISSN 2786-9024

політехніка». Серія «Інформаційні технології та засоби обчислювальної техніки», 287, 123-134, 2021.

[4] A.Gupta, S.Sharma, "Smart governance for sustainable development: A review of the literature", *International Journal of Sustainable Development & World Ecology*, vol. 29, iss. 2, 173-182, 2022.

[5] Z.Pang, Y.Wang, Y.Zhang, "Smart city governance for sustainability: A review of the literature", *Sustainability*, vol. 14, iss. 15, 6970, 2022.

[6] Індекс цифрової економіки та суспільства: прогрес ЄС. [Онлайн]. URL: http://surl.li/rlcqv. Дата звернення: 06.01.2024.

[7] Програма діяльності Кабінету Міністрів України: Постанова Кабінету Міністрів України від 12.06.2020 р. № 471. [Онлайн]. URL: http://surl.li/rlcrp. Дата звернення: 06.01.2024.

[8] Національна стратегія із створення безбар'єрного простору в Україні на період до 2030 року: Розпорядження Кабінету Міністрів України від 14.04.2021 р. № 366-р. [Онлайн]. URL: http://surl.li/dhqql. Дата звернення: 06.01.2024.

[9] Державна стратегія регіонального розвитку на 2021-2027 роки : постанова від 05.08.2020 р. № 695. [Онлайн]. URL: http://surl.li/cczhd. Дата звернення: 06.01.2024.

[10] Стратегія реформування державного управління на 2022-2025 роки: Розпорядження Кабінету Міністрів України від 21.07.2021 р. № 831-р. [Онлайн]. URL: http://surl.li/rlcvv. Дата звернення: 06.01.2024.

[11] План дій із впровадження Ініціативи «Партнерство «Відкритий Уряд» у 2021-2022 роках: Розпорядження Кабінету Міністрів України від 21.02.2021 р. № 149-р. [Онлайн]. URL: http://surl.li/rlcwi. Дата звернення: 06.01.2024.

[12] Деякі питання цифрової трансформації: Розпорядження Кабінету Міністрів України від 17.02.2021 р. № 365-р. [Онлайн]. URL: http://surl.li/rlcxw. Дата звернення: 06.01.2024.

[13] Концепція розвитку системи електронних послуг в Україні: Розпорядження Кабінету Міністрів України від 16.11.2016 р. № 918-р. [Онлайн]. URL: http://surl.li/rlcyv. Дата звернення: 06.01.2024.

[14] Концепція розвитку штучного інтелекту в Україні: Розпорядження Кабінету Міністрів України від 02.12.2020 р. № 1556-р. [Онлайн]. URL: http://surl.li/qmhdv. Дата звернення: 06.01.2024.

[15] Стратегія розвитку сфери інноваційної діяльності на період до 2030 року: Розпорядження Кабінету Міністрів України від 10.07.2019 р. № 526-р. [Онлайн]. URL: http://surl.li/rldae. Дата звернення: 06.01.2024.

[16] Дорожня карта реформування ІТ-освіти: наказ Міністерства освіти і науки України, Міністерства цифрової трансформації України від 23 грудня 2021 р. № 1418/181.

[17] План заходів з реалізації Концепції розвитку штучного інтелекту в Україні на 2021-2024 роки: Розпорядження Кабінету Міністрів України від 21.05.2021 р. № 438-р. [Онлайн]. URL: http://surl.li/rlday. Дата звернення: 06.01.2024.

[18] Стратегія кібербезпеки України: Указ Президента України від 26.08.2021 р. № 447/2021. [Онлайн]. URL: http://surl.li/bnpdr. Дата звернення: 06.01.2024.

[19] Концепція забезпечення національної системи стійкості: Указ Президента України від 27 вересня 2021 р. № 479/2021. [Онлайн]. URL: http://surl.li/kmjwd. Дата звернення: 06.01.2024.

CONCEPTUAL FOUNDATIONS OF GLOBAL RESILIENCE OF A SMART STATE

Iaroslav Dorohyi, Iryna Berdychenko

This article is dedicated to the examination of the conceptual foundations of the global resilience of a smart state. The authors explore the contemporary dimension of Ukraine's development in the context of the integration of technologies and innovations into all aspects of life. In particular, the article investigates the impact of information technologies, artificial intelligence, and other modern technologies on the efficiency of governance, socio-economic development, and ecological resilience.

The article provides a detailed analysis of key aspects of the creation and functioning of smart states, including the role of digital infrastructures, open data, and e-government. The authors analyze the principles of interaction between a smart state and citizens, businesses, and other members of society to achieve a high level of resilience and development.

The article also examines challenges and risks associated with the implementation of modern technologies in the state system and proposes strategies for ensuring cybersecurity and privacy protection in the digital environment. Special attention is given to the ideas of resilience development and ethical aspects of technology use to achieve global resilience of a smart state, understanding such definitions as digital resilience and global resilience, and accordingly, determining their structure and patterns of use

Keywords: *global resilience, smart state, resilience, digital infrastructure, sustainability, conceptual foundations*

REFERENCES

[1] J.Dupont. The Smart State. Redesigning government in the era of intelligent services. London: Policy Exchange 8 – 10 Great George Street, 2018. 45 p. [Online]. URL: http://surl.li/rlcqa. Accessed: 06.01.2024.

[2] I.V.Vasylieva, O.Iu.Hubenko, "Kiberbezpeka yak skladova hlobalnoi stiikosti smart-derzhavy", *Naukovi pratsi Natsionalnoho universytetu «Lvivska politekhnika». Seriia «Informatsiini tekhnolohii ta zasoby obchysliuvalnoi tekhniky»*, 293, 121-132, 2022.

[3] Yu.O.Hubenko, I.V.Vasylieva, "Etychni aspekty vykorystannia tekhnolohii u formuvanni smart-derzhavy", *Naukovi pratsi Natsionalnoho universytetu «Lvivska politekhnika». Seriia «Informatsiini tekhnolohii ta zasoby obchysliuvalnoi tekhniky»*, 287, 123-134, 2021.

[4] A.Gupta, S.Sharma, "Smart governance for sustainable development: A review of the literature", *International*

Journal of Sustainable Development & World Ecology, vol. 29, iss. 2, 173-182, 2022.

[5] Z.Pang, Y.Wang, Y.Zhang, “Smart city governance for sustainability: A review of the literature”, *Sustainability*, vol. 14, iss. 15, 6970, 2022.

[6] Indeks tsyfrovoi ekonomiky ta suspilstva: prohres YeS. [Online]. URL: http://surl.li/rlcqv. Accessed: 06.01.2024. (In Ukranian).

[7] Prohrama diialnosti Kabinetu Ministriv Ukrainy: Postanova Kabinetu Ministriv Ukrainy vid 12.06.2020 r. № 471. [Online]. URL: http://surl.li/rlcrp. Accessed: 06.01.2024. (In Ukranian).

[8] Natsionalna stratehiia iz stvorennia bezbariernoho prostoru v Ukraini na period do 2030 roku: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 14.04.2021 p. № 366-p. [Online]. URL: http://surl.li/dhqql. Accessed: 06.01.2024. (In Ukranian).

[9] Derzhavna stratehiia rehionalnoho rozvytku na 2021-2027 roky : postanova vid 05.08.2020 r. № 695. [Online]. URL: http://surl.li/cczhd. Accessed: 06.01.2024. (In Ukranian).

[10] Stratehiia reformuvannia derzhavnoho upravlinnia na 2022-2025 roky: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 21.07.2021 r. № 831-p. [Online]. URL: http://surl.li/rlcvv. Accessed: 06.01.2024. (In Ukranian).

[11] Plan dii iz vprovadzhennia Initsiatyvy «Partnerstvo «Vidkrytyi Uriad» u 2021-2022 rokakh: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 21.02.2021 p. № 149-p. [Online]. URL: http://surl.li/rlcwi. Accessed: 06.01.2024. (In Ukranian).

[12] Deiaki pytannia tsyfrovoi transformatsii: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 17.02.2021 r. № 365-p. [Online]. URL: http://surl.li/rlcxw. Accessed: 06.01.2024. (In Ukranian).

[13] Kontseptsiia rozvytku systemy elektronnykh posluh v Ukraini: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 16.11.2016 r. № 918-p. [Online]. URL: http://surl.li/rlcyv. Accessed: 06.01.2024. (In Ukranian).

[14] Kontseptsiia rozvytku shtuchnoho intelektu v Ukraini: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 02.12.2020 r. № 1556-p. [Online]. URL: http://surl.li/qmhdv. Accessed: 06.01.2024. (In Ukranian).

[15] Stratehiia rozvytku sfery innovatsiinoi diialnosti na period do 2030 roku: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 10.07.2019 r. № 526-p. [Online]. URL: http://surl.li/rldae. Accessed: 06.01.2024. (In Ukranian).

[16] Dorozhnia karta reformuvannia IT-osvity: nakaz Ministerstva osvity i nauky Ukrainy, Ministerstva tsyfrovoi transformatsii Ukrainy vid 23 hrudnia 2021 r. № 1418/181.

[17] Plan zakhodiv z realizatsii Kontseptsii rozvytku shtuchnoho intelektu v Ukraini na 2021-2024 roky: Rozporiadzhennia Kabinetu Ministriv Ukrainy vid 21.05.2021 r. № 438-p. Accessed: 06.01.2024. [Online]. URL: http://surl.li/rlday. (In Ukranian).

[18] Stratehiia kiberbezpeky Ukrainy: Ukaz Prezydenta Ukrainy vid 26.08.2021 r. № 447/2021. Accessed: 06.01.2024. [Online]. URL: http://surl.li/bnpdr. (In Ukranian).

[19] Kontseptsiia zabezpechennia natsionalnoi systemy stiikosti: Ukaz Prezydenta Ukrainy vid 27 veresnia 2021 r. № 479/2021. [Online]. URL: http://surl.li/kmjwd. Accessed: 06.01.2024. (In Ukranian).

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10804951

Інформаційна безпека

БЕЗПЕКА ТА ЗАХИСТ НАВЧАЛЬНИХ LMC СИСТЕМ

Н.О. Маслова[1], О.М. Любименко[1]

[1] Department of Applied Mathematics and Informatics, Donetsk National Technical University, Lutsk, Ukraine
E-mail: olena.liubymenko@donntu.edu.ua

Отримано 31.12.2023
Прийнято до публікації 19.01.2024
Опубліковано 01.04.2024

АНОТАЦІЯ

Інформаційні технології відіграють значну роль у навчальному процесі й забезпеченні якісних результатів навчання в умовах дистанційної освіти. Сучасною тенденцією є розвиток інтерактивних систем, які включають елементи аудіо- та відео- матеріалів, графіки, презентацій, інтернет посилань, матеріали з різних джерел та у різних форматах. Додавання інтерактивного контенту до в навчальних систем підвищує ризики інформаційної безпеки сучасних системах LMC. Акцент в роботі зроблено на необхідності забезпечення захисту програмного забезпечення систем, розміщеного в них освітнього контенту, результатів навчання та персональних даних учасників. Створюючи захист, слід звертати увагу на потенційні загрози, такі як несанкціонований доступ, недосконале програмне забезпечення, неякісні плагіни оновлення, копіювання матеріалів, шахрайство та кібератаки. Проаналізовано вразливості найбільш поширених сучасних LMC систем. Приділено увагу можливостям використання хмарних технологій для розміщення модулів систем, навчальних матеріалів й результатів навчання на хмарних сервісах. Застосування хмар гарантує постійний доступ до навчальної системи, надійне зберігання матеріалів, додатковий захист конфіденційної та персональної інформації. Тому для забезпечення ефективного та безпечного навчання з урахуванням сучасних інтерактивних підходів до подання учбових матеріалів, важливо розвивати системи та застосовувати технології, які забезпечують постійний доступ до освітнього контенту, надійний захист даних (у тому числі й персональних), збереження цілісності інформації та дотримання принципів конфіденційності.

***Ключові слова**: системи LMC, інтерактивне навчання, уразливості систем дистанційного навчання, захист, хмарні технології, конфіденційність, доступність, цілісність, надійність, інформація, персональні дані, освітній контент.*

ВСТУП

Бурхливий розвиток інформаційних технологій сприяв становленню багатьох сучасних технологій, які застосовуються у виробництві, медицині, навчанні. ІТ-технології, штучний інтелект, інтерактивні методи навчання є звичною дійсністю, без якої складно уявити процес надання знань й сьогодні дистанційні системи є невід'ємною частиною учбового процесу. Ці системи не тільки забезпечують доступ до знань у будь-якій точці світу, але й створюють унікальні можливості для взаємодії, адаптації до індивідуальних потреб та розвитку навчального процесу в інтерактивному ключі.

При експлуатації систем постійно зростають вимоги до технічних засобів й їх оновлення в процесі експлуатації, програмного забезпечення, яке не є простим в застосуванні та обслуговуванні, до створення й зберігання навчального контенту, який є однією з складових частин навчальної системи. Бази даних вимагають кваліфікованого супроводу, поповнення, розвитку, що складно здійснювати у разі великої кількості користувачів й мінімальної кількості (як правило) супроводжуючого персоналу.

Одночасно зі зростанням популярності й розповсюдженості дистанційної форми навчання, зростає розуміння у необхідності надійного захисту навчальних систем, приділяється увага захищеності контенту, програмного забезпечення, персональних даних учасників учбового процесу.

Навчальні системи мають конфіденційну інформацію, таку як особисті дані студентів та викладачів. А тому недостатні заходи безпеки можуть призвести до:

- несанкціонованого доступу до інформації, що зафіксована в системах;
- копіювання матеріалів, розміщених в навчальних системах, порушення авторських прав;
- шахрайства під час проведення тестів та іспитів (підміна користувача);
- ураження й порушення цілісності даних при проведенні кібератак та порушень безпеки;
- несанкціонованого доступу до навчальних матеріалів або систем, що може порушити конфіденційність даних.

Ці проблеми стають все більш актуальними в контексті розвитку електронної та дистанційної освіти. Одна з основних проблем використання систем навчання полягає в недостатньому рівні захисту систем та контенту, що в них розміщується та застосовується, особливо в умовах, коли кількість загроз постійно зростає.

АНАЛІЗ ЛІТЕРАТУРНИХ ДАНИХ ТА ПОСТАНОВКА ПРОБЛЕМИ

З розвитком технологій віртуальної та доповненої реальності, штучного інтелекту та онлайн-платформ, інтерактивні системи стали засобом для ефективного навчання. Інтерактивні системи являють собою інноваційні засоби, для їх створення використовують сучасні технології динамічного та захопливого освітнього досвіду.

Засоби проведення інтерактивного навчання розподіляють на чотири великі групи [1]. Це базові платформи для створення онлайн-курсів (Moodle, Canvas, Blackboard, Schoology), середовища віртуальної та доповненої реальності (Unity3D, A-Frame, ARCore та ARKit), системи для створення освітнього контенту з елементами гейміфікації (Kahoot!, Classcraft, Quizziz), інструменти для створення інтерактивних мультимедійних матеріалів (Adobe Captivate, Articulate Storyline) та спеціалізовані адаптивні освітні платформи (Smart Sparrow, Microsoft Teams, Knewton, Edpuzzle, Google Classroom), які дозволяють застосовувати інтерактивний контент, включати його в освітній контент.

Перелічені системи включають поширені й унікальні функції, побудовані з застосуванням різноманітних підходів до навчання, їх вибір залежить від конкретних потреб та цілей навчального процесу. Ці інструменти надають викладачам і розробникам освітніх матеріалів можливість створення інтерактивних систем, що сприяють активній участі студентів і підвищенню ефективності навчання. По-суті, це LMS (Learning Management System) системи, системи управління навчанням, в яких для підвищення ефективності навчання застосовують нестандартні форми представлення знань - ігри, графічні редактори, CAD-CAM системи, інтегровані середовища, аудіо та відео-записи й багато іншого. Це програмне забезпечення, найчастіше хмарне, яке дає змогу створювати освітні продукти в електронному вигляді та організовувати онлайн-заняття, включати в них інтерактивні елементи.

Галузь дистанційного навчання є на даний момент достатньо розвиненою та сталою. Відомими й широко застосовними є системи Moodle, Canvas, Webtutor, ILIAS та інші. Але під час створення цих систем проблеми захисту контенту, персональних даних, програмного забезпечення не були оголошені як першочергові. Це привело до необхідності розв’язування питань захисту існуючих систем в умовах їх активної експлуатації.

Перші застереження й спроби дослідження необхідності захисту систем дистанційного навчання виникли у першому десятиріччі поточного століття. Так, у роботі [2] автори наголосили на недостатній увазі, яку приділяють розробники навчальних систем до питань безпеки своїх продуктів. У роботі [3] проведено аналіз та класифікацію загроз порушення безпеки систем дистанційного навчання. Автори зробили висновок, що незалежно від архітектури системи, слід виділити дві групи загроз: загальні та специфічні.

До загальних автори віднесли загрози доступності, dos-атаки, проблеми переповнення буферів, впливу SQL-ін’єкцій, спроби підбору паролів, та інші, характерні для автоматизованих інформаційних систем взагалі. Ці загрози достатньо відомі й можуть бути нейтралізовані використанням методів та засобів захисту інформації загального призначення. А в якості специфічних для

ISSN 2786-9024

систем дистанційного навчання автори називають загрози, які зумовлені взаємодією суб’єктів та об’єктів навчального процесу.

Частково проблеми захищеності розглянуті й в інших дослідженнях. Так, в [4] автори роблять акцент на організаційно-адміністративних методах забезпечення захисту однієї з найбільш популярних систем дистанційного навчання – системі Moodle. У роботі [5] показано особливості забезпечення захисту інформації, яка міститься в системі дистанційного навчання Moodle й розглянуто основні характеристики операційної системи CentOS, яка застосовується для забезпечення навісного захисту системи, показано особливості проведення захисту інформації при роботі з цією операційною системою. У роботі [6] увага акцентована на необхідності застосування систем шифрування в процесі отримання та зберігання даних.

Робота [7] наближена до сучасних вимог та стандартів захисту інформаційних систем. В ній детально проаналізовані базові проблеми захисту інформації в сучасних системах дистанційного навчання та загрози з точки зору інформаційної безпеки для таких систем, перелічені основні цілі, які може переслідувати зловмисник при реалізації атак на системи дистанційного навчання (СДН) та уразливості через які він здійснює атаки. Здійснено порівняння найбільш поширених LMC за такими ключовими параметрами, як загрози хибної реєстрації та автентифікації, загрози порушення достовірності результатів контролю знань та загрози впровадження шкідливого програмного забезпечення. Основну увагу приділено підходам до захисту СДН від загроз підміни користувача, загроз використання програмних ботів і скриптів, а також загроз використання лекцій, електронних довідників та інших сторонніх навчальних матеріалів. Запропоновано механізм захисту від загроз, й автори наголошують, що алгоритм дій може бути використаний у будь-якій системі дистанційного навчання для захисту від загроз порушення достовірності знань.

У [8] показано, що найбільш уразливими з точки зору інформаційної безпеки є процеси:

- передачі ідентифікаційних і аутентифікаційних даних користувача;
- обмін даними між браузером віддаленого користувача і веб-сайтом навчальної системи;
- авторизації користувача (на сервері системи дистанційного навчання і в інформаційно- комунікаційній системі навчального закладу);
- витяг і запис даних в бази навчального закладу;
- обмін даними між сервером СДН і сервером ІС навчального закладу.

Вказано, що зловмисник може бути як зовнішнім, так і внутрішнім, перелічені цілі, котрі він переслідує та уразливості, які застосовує. Це, зокрема – уразливості в веб-додатку і сервісах СДН; слабкі паролі і недоліки процесу автентифікації користувачів на сервері СДН; помилки в конфігурації і адмініструванні СДН; шкідливе програмне забезпечення (віруси, троянські програми, програмні бомби і закладки); слабкості системи захисту інформації.

Таким чином, до основних проблем захисту електронних навчальних систем слід віднести:

- недостатню захищеність від несанкціонованого доступу;
- ризики несанкціонованого копіювання та плагіату;
- порушення достовірності результатів контролю знань;
- безпеку даних під час розміщення та передачі даних;
- проблеми управління доступом;
- порушення конфіденційності.

Одним із актуальних шляхів розвитку освітніх систем є використання хмарних технологій. Хмарні технології є надзвичайно популярними, кількість їх застосувань у різних сферах стрімко зростає. Їх впровадженню й активному розповсюдженню сприяють такі фактори, як:

1. можливість постійного доступу до навчального контенту;
2. збереження матеріалів та результатів навчання на віддалених серверах в центрах обробки, завдяки чому з користувача знімається проблема обслуговування технічної частини навчальної системи;
3. наявність засобів безпеки при передачі, прийманні та збереженні інформації, що гарантується самою технологією хмарних структур;
4. додаткова ідентифікація користувачів в процесі отримання доступу до хмарного ресурсу з фіксацією й контролем адреси входу.

Проблеми захисту неможливо розв’язати без адаптації архітектури та конфігурації навчальних систем до існуючих технологій. Тому слід приділити увагу не тільки проблемам й особливостям так званого навісного захисту існуючих систем, а й дослідженню можливостей зниження ризиків інформаційної безпеки при експлуатації навчальних LMC систем при активному застосуванні сучасних хмарних технологій. Й об’єднати фактори гарантованої працездатності апаратного забезпечення, постійної доступності, сертифікованої захищеності, можливостей відокремленого збереження,

послуг копіювання даних й антивірусного захисту, які надають хмарні сервіси з перевагами LMC систем в сфері надання освітніх послуг в сучасних реаліях.

Враховуючи вищенаведене, об'єктом дослідження є методології захисту LMC систем й засобів створення учбового контенту та навчального електронного ресурсу з огляду на сучасні погляди забезпечення безпеки учасників.

Предметом дослідження є вразливості навчального середовища при взаємодії користувача із інтерактивними та навчальними ресурсами.

Метою роботи є дослідження проблем захищеності сучасних LMC систем з урахуванням вразливостей програмного забезпечення, навчального контенту, результатів навчання та персональних даних учасників освітнього процесу, вплив хмарних технологій на безпеку систем, а також надання рекомендацій для підвищення захищеності навчальних електронних ресурсів.

Для досягнення мети необхідно виконати наступні задачі:

- виявити та проаналізувати загальні проблеми захисту систем LMC з урахуванням сучасних загроз інформаційній безпеці;
- обрати найбільш поширені системи дистанційної освіти й системи керування навчанням;
- дослідити вразливості поширених систем LMC й можливості застосування сучасних інформаційних технологій, зокрема, хмарних, для зниження ризиків порушення безпеки.

МАТЕРІАЛИ ТА МЕТОДИ ДОСЛІДЖЕНЬ

В роботі використано методи структурного і порівняльного аналізу, з інформаційним і аналітичним підходом розглянуто наукову та методичну літературу, а також онлайн ресурси.

РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

А. ДОСЛІДЖЕННЯ СИСТЕМ LMC ЗА ПАРАМЕТРАМИ БЕЗПЕКИ

У 2024 році сайт ahaslides.com [9] опублікував огляд, у якому проаналізовані сучасні системи LMS. Розглянемо п'ять з них у пропонованій в статті послідовності. Доповнимо огляд платформою інтеграції Edpuzzle, яка дозволяє об'єднувати популярні рішення. Й проаналізуємо рішення щодо захищеності й інформаційної безпеки ресурсів, на яких наголошують розробники на відповідних сайтах.

Називаючи найбільш поширеною та застосовною систему Google Classroom, автори, тим, не менш, перелічили її основні недоліки: обмежена інтеграція з іншими програмами, відсутність розширених (спеціалізованих для організації класу) функцій, необхідність конвертації своїх файлів у формат Google, відсутність автоматизованих вікторин чи тестів, вікові обмеження (студенти до 13 років можуть використовувати Classroom лише з обліковими записами Google Workspace for Education або Workspace for Nonprofits) й, найголовніший для даного дослідження недолік - Google відстежує поведінку користувачів і дозволяє рекламу на їхніх сайтах, що слід ідентифікувати як порушення конфіденційності.

Google Classroom, як і інші продукти Google, розміщена на хмарі. Google використовує свої власні хмарні інфраструктури, відомі як Google Cloud Platform, для забезпечення інфраструктури, потрібної для функціонування Google Classroom. Дані, включаючи інформацію про користувачів, завдання, матеріали та інше, зберігаються на серверах Google, які розташовані у різних частинах світу.

Іншою сучасною системою є платформа інтеграції систем Edpuzzle [9]. Edpuzzle – це проста у використанні, сучасна платформа, яка працює з відео- аудіо, текстовими, презентаційними матеріалами й дозволяє додавати до них власні запитання або записи, контролювати й керувати процесом навчання, відслідковувати прогрес учнів.

Edpuzzle [10] пропонує об'єднання на спеціалізованій платформі декількох існуючих систем дистанційного навчання з метою уніфікації застосування різнорідних систем, взаємообміну даними між ними, допомагає заощадити час і зусилля на експорт/імпорт даних між різними системами, завдяки єдиному інтерфейсу спрощує роботу з системою. Викладач може підключити відому йому й наповнену систему до Edpuzzle й транслювати завдання через новий, сучасний засіб, перенести контент в нове навчальне середовище, об'єднати декілька блоків навчальних матеріалів при їх попередньому розміщенні на різних ресурсах в єдине ціле.

Зараз Edpuzzle інтегрується з системами Google Classroom, Microsoft Teams, Canvas, Schoology, Moodle, Blackboard, Blackbaud, PowerSchool, Clever, D2L Brightspace.

Edpuzzle – достатньо безпечне середовище, яке дозволяє будувати повноцінні, сучасні навчальні курси, з застосуванням звичних користувачам середовищ. Edpuzzle серйозно ставиться до безпеки та конфіденційності. Ресурс нагороджено кількома сертифікатами безпеки та відповідності вимогам захисту конфіденційності. Edpuzzle - може бути розміщена на

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10804951

хмарних серверах, однак недостатня захищеність приєднуваної системи може стати загрозою безпеці всього інтегрованого середовища.

Canvas LMS (https://www.instructure.com/canvas) – це хмарно - орієнтована система керування навчанням. Він відомий своїм зручним інтерфейсом, надійністю та повним набором функцій, призначених для зручного викладання і навчання.

Безпека вбудована безпосередньо в хмарну платформу, інфраструктуру та процеси, відповідно до стандартів SOC 2 і ISO 27001. Дані розміщуються в хмарі Instructure та доставляються через Amazon Web Services (AWS). Користувачі можуть входити за допомогою облікових даних облікового запису Google Cloud, використовуючи мову розмітки декларації безпеки (SAML). Canvas гарантує 99.99% часу безперебійної роботи і цілодобове функціонування платформи для всіх користувачів, тож Canvas вважається однією з надійних LMS.

Edmodo (офіційна web-сторінка - https://www.edmodo.com) - це освітній сайт, який являє собою усічену соціальну мережу за типом Facebook, яка дозволяє спілкуватися вчителям та учням, об'єднавшись навколо процесу навчання у школі.

Edmodo може бути розміщена на хмарних серверах. Використання хмарної інфраструктури дозволить Edmodo швидко впроваджувати оновлення та нові функції, забезпечуючи користувачам доступ до оновлень без необхідності вручну оновлювати програмне забезпечення. Крім того, хмарна інфраструктура може забезпечити резервне копіювання даних й покращити захист та відновлення даних в разі виникнення проблем.

Edmodo – це захищена освітня мережа, яка наголошує на суворому контролі безпеки для повідомлень і для всіх інших комунікацій на платформі.

Moodle - одна з найпопулярніших систем управління навчанням у світі, є модульним об’єктно-орієнтованим динамічним навчальним середовищем з відкритим вихідним кодом, система керування навчанням (LMS). Система Для створення персоналізованого навчального простору, Moodle пропонує власну послугу хостингу MoodleCloud. Хоча Moodle вважається достатньо надійною та захищеною для користувачів та контенту, захист інформації щодо конфіденційності, цілісності та доступності потребує використання різних методів та заходів[11].

Щоб забезпечити роботу Moodle, потрібні три складові: веб-сервер, база даних і поштовий сервер. Хостинги, які використовуються для розміщення серверів Moodle, повинні бути обладнані автоматизованими системами захисту від DDOS-атак та антивірусним програмним забезпеченням для захисту програмних файлів від різних видів порушень безпеки. [12].

Окрім того з метою захисту інформаційних ресурсів в системі передбачено:

- використання паролів для доступу до інформаційних ресурсів;
- політики розмежування прав доступу, що дозволяє призначати різний рівень повноважень для студентів, викладачів та адміністраторів;
- автоматизація виконання системних дій, включаючи доступ до тестів та обмеження настроювань без адміністративних прав;
- створення резервних копій системи;
- IP-блокатори для перевірки вхідних Інтернет-адрес і блокування небажаних IP-адрес;
- наявність безпечного http-з’єднання для сторінок входу до системи;
- вбудований антивірус «Clam AV», для перевірки всіх завантажених файлів та навчальних матеріалів на наявність вірусів;
- налаштування показу особистих даних користувачів (уподобання);
- комплексний захист інформації в базі даних, який забезпечує обмежений доступ до неї та розміщення інформації для тестів в різних таблицях..

В системі управління навчанням «Moodle» в будь-якому освітньому закладі використовується інформаційне середовище, що є сукупністю навчальних матеріалів, засобів підтримки навчального процесу, представлених в електронному вигляді, а також різні засоби, методи та форми комунікації між суб’єктами освітнього процесу.

Таким чином, можна наголосити, що система дистанційного навчання «Moodle» має достатній рівень захищеності інформації. На сайті [13] розміщена велика кількість інформаційно-довідникових матеріалів щодо організації й особливостей побудови безпеки системи Moodle. Але вразливим місцем системи вважаються слабкі паролі (пароль може обрати або змінити користувач), невірна конфігурація доступу або недостатня організація користувацьких прав. Це може створювати ризики для безпеки даних у Moodle. Система не оновлюється автоматично. Несвоєчасне встановлення плагінів можуть бути використані зловмисниками для отримання доступу до системи, внесення змін у інформацію або виконання інших злочинних дій. Крім того, широкі можливості застосування в якості навчальних матеріалів різноманітного контенту (відео, презентації, інтернет-посилання), не перевіреного або недостатньо перевіреного з точки зору безпеки контенту також можуть послабити захист системи.

 https://doi.org/10.5281/zenodo.10804951

Moodle може бути розміщена як у хмарі, так і на локальних серверах, в залежності від вибору організації та її потреб у забезпеченні безпеки, доступності та масштабованості.

AhaSlides (https://ahaslides.com/ru/features/) – хмарна платформа, яка дає змогу презентувати та проводити інтерактивні заходи. Вона вважається аналогом PowerPoint.

У політиці безпеки AhaSlides реалізовано вимоги конфіденційності, управління доступом та контролю користувача. Резервні копії даних розміщені на платформі Amazon Web Services, яка відповідає стандартам ISO/IEC 27001:2013, 27017:2015 та 27018:2014, сертифікована як постачальник послуг PCI DSS 3.2 рівня 1 і проходить SOC 1, SOC 2 та SOC3. Копії зберігаються на Amazon RDS з використанням повного диска, стандартного шифрування AES ARS з унікальним ключем для кожного сервера. Файлові вкладення у презентації AhaSlides зберігаються у службі Amazon S3 з унікальними посиланнями, доступними через захищене з'єднання HTTPS.

AhaSlides використовує промисловий стандарт безпеки транспортного рівня (TLS) зі 128-бітним шифруванням AES для всіх з'єднань, а паролі хешуються та шифруються за допомогою алгоритму PBKDF2 (з SHA512).

На сайті опубліковано план перегляду безпеки та процес управління інцидентами для виявлення та реагування на них. Платформа дотримується сучасних стандартів безпеки та відкрита для перевірки.

AhaSlides може бути розміщена на хмарних серверах для швидкого впровадження оновлень та резервного копіювання даних.

Microsoft Teams називають робочим середовищем для спільної роботи.

Microsoft Teams використовує хмарну інфраструктуру Microsoft Azure. Це означає, що дані, повідомлення, файли, календарі, відеоконференції та інші, зберігаються на серверах Microsoft у різних місцях світу. Що підвищує їх надійність.

Використання хмарної інфраструктури дозволяє забезпечити високу доступність, масштабованість та безпеку продукту, оскільки Microsoft має великий досвід у сфері хмарних технологій і активно вдосконалює заходи безпеки. Ці заходи включають шифрування даних у спокої, заходи фізичної та мережевої безпеки, а також системи виявлення та запобігання вторгненням.

Недоліком Teams є підвищені ризики небезпеки, пов’язані з тим, що кожен може створити команду або вільно завантажувати на канал файли з конфіденційною інформацією [10]. Крім того, невірна настройка конфігурації Teams, помилки у встановлені налаштування конфіденційності чи доступу, може призвести до небажаних витоків інформації або несанкціонованого доступу.

Б. СИСТЕМАТИЗАЦІЯ РЕЗУЛЬТАТІВ

В огляді [9] пропонуються й інші системи, як то Classcraft та Excalidraw. Але вони не є повноцінними навчальними середовищами, а тільки варіаціями інструментів, які можуть бути застосовані для організації дистанційного навчання, й можуть впливати на загальну захищеність навчального середовища при їх застосуванні, але у цій статті не розглядаються.

За даними [9], найбільша кількість вразливостей, які використовуються зловмисниками при атаках з зовнішньої мережі на LMC системи, виявлена в прикладних програмах, що активно застосовуються при підготовці матеріалів для наповнення систем дистанційного навчання. Ці програми включають в себе браузери, які використовуються користувачами СДН для доступу до веб-сайтів, а також Adobe Reader, Adobe Flash Player і Oracle Java, які використовуються для виконання скриптів і обробки документів та мультимедійних файлів.

Систематизуємо проблеми можливих порушень безпеки систем LMC (таблиця 1) і запропонуємо заходи захисту від найбільш активних погроз безпеки.

Таблиця 1. Проблеми захисту LMC - систем

№	Вразливості	LMC- системи						
		Google Classroom	*Edpuzzle*	*Canvas*	*Edmodo*	*Moodle*	*AhaSlides*	*Teams*
1	Несанкціонований доступ до акаунтів	+	+	+	+	+	+	+
2	Витік конфіденційної інформації	+		+				
3	Віруси та шкідливе програмне забезпечення	+				+		+
4	Неякісні практики захисту даних	+	+	+	+	+	+	+
5	Атаки з перехопленням (слабке шифрування)	+						
6	Вразливості програмного забезпечення		+	+	+	+	+	+
7	Застарілі версії			+		+	+	

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10804951

8	Помилки в конфігурації							+
9	Недостатній контроль доступу		+	+	+		+	
10	Втрати даних		+	+	+		+	
11	Порушення правил безпеки користувачами					+	+	

Наведемо розшифровку вразливостей, перелічених в таблиці.

Несанкціонований доступ до акаунтів може включати вторгнення в акаунти користувачів, використання слабких паролів або злам паролів, а також атаки фішингу. Несанкціонований доступ може призвести до витоку конфіденційної інформації, такої як персональні дані учнів та викладачів.

Якщо дані захищено недостатньо й вони потрапляють до неправомірних рук, це може призвести до *витоку особистої інформації*, оцінок, завдань або іншої конфіденційної інформації.

Шкідливе програмне забезпечення може заражати пристрої користувачів, які використовують LMC – системи, поширювати конфіденційні дані або завдати шкоди комп’ютерам користувачів.

Під неякісними практиками захисту даних розуміємо недостатні заходи безпеки, такі як слабкі паролі, використання незахищених мереж Wi-Fi, недостатня організація користувацьких прав або невірна обробка конфіденційної інформації, які можуть підвищити ризик порушення безпеки.

Атаки з перехопленням. Можливість їх реалізації створюються, якщо дані, що передаються від користувача в систему (або канали передачі даних) не мають належного шифрування. Це може створити ризик перехоплення чутливої інформації в мережі.

Вразливості програмного забезпечення – це помилки в самих системах (характерно для нових або «свіже оновлених» розробок) й недостатньо перевірених плагінах можуть бути використані зловмисниками для отримання доступу до системи, внесення змін у інформацію або виконання злочинних дій.

Відсутність автоматичного оновлення систем, використання застарілих версій систем, або невчасне встановлення патчів можуть залишити систему вразливою до відомих атак та загроз безпеки.

Помилки в конфігурації, невірно встановлені налаштування конфіденційності чи доступу, може призвести до небажаних витоків інформації

+ *Неадекватно налаштований контроль доступу* до курсів та матеріалів може призвести до неправомірного доступу до конфіденційної інформації.

Втрати даних - непередбачені ситуації, такі як вірусні атаки або технічні проблеми, можуть призвести до втрати або пошкодження даних, які зберігаються в системі.

Таким чином, проблема безпеки LMC систем э комплексною та багатоаспектною задачею.

Заходи з захисту систем LMC (Language Model-based Conversational systems) повинні включати наступні процедури.

1. *Обов’язкове проведення аутентифікації та авторизації*. Слід відслідковувати й забезпечувати механізми аутентифікації користувачів, щоб вони могли взаємодіяти з системою лише після підтвердження своєї ідентичності. Права доступу користувачів повинні бути налаштовані таким чином, щоб забезпечувати обмеження доступу до адміністративного функціоналу системи.

2. *Регулярне оновлення програмного забезпечення*, що допоможе виявленню та усуненню вразливостей, які можуть бути застосовані зловмисниками для порушення безпеки LMC системи.

3. *Регулярні резервні копії* даних запобігають втраті інформації в разі кібератаки або технічних проблем.

4. *Шифрування даних*. Слід використовувати алгоритми надійного шифрування для захисту конфіденційної інформації, яка передається між користувачем і системою, а також для зберігання даних на сервері.

5. *Дотримання вимог щодо захисту* особистих даних користувачів та вимог законодавства щодо захисту даних дозволить запобігти витоку персональних даних.

6. *Захист від кібератак*. Рекомендується вживайте різноманітні заходи безпеки, такі як використання мережевих брандмауерів, систем виявлення вторгнень (IDS), і системи запобігання вторгненням (IPS) для захисту системи від різних видів кібератак, таких як DDoS або SQL ін'єкції.

7. *Аудит безпеки*. Регулярні аудити безпеки дозволять виявляти потенційні вразливості системи та вживати заході для їх усунення.

8. *Інструктажі користувачів систем*. Цей захід допоможе залучити учасників навчального процесу до розпізнавання потенційних загроз безпеки та активно реагувати на них. Тренінги з кібербезпеки слід проводити для всіх учасників, які користуються системою LMC.

9. *Доступ до даних слід обмежити* діючою політикою та розробленою й функціонуючою системою доступу.

Це загальні рекомендації. Їх реалізація для користувачів системи значно спрощується у випадку розміщення LMC системи на хмарних хостингах. Хмарні технології дозволяють вирішити практично всі вищезазначені проблеми та забезпечують захист, надійність, доступність та конфіденційність усіх складових складних і спеціалізованих структур, якими є навчальні системи та сучасні модулі управління освітнім контентом.

Аутентифікація й авторизація, створення резервних копій даних, налагодження автоматичного оновлення програмного забезпечення систем, стандартне, високоякісне шифрування даних, які зберігаються, накопичуються та обробляються в системі, шифрування даних при прийманні й передаванні, антивірусний контроль з залученням міжнародно сертифікованих пакетів, це лише короткий перелік переваг розміщення навчальних систем на хмарах. Тож першим питанням навчального закладу при обранні системи для організації навчального процесу повинно бути питання безпеки й пошук ресурсу, який цю безпеку гарантує.

Таким чином, застосування хмарних технологій в освітніх системах сприяє покращенню процесу навчання, захисту накопиченої в системах інформації, доступності освітніх заходів.

А способи розподілу ресурсів LMC систем з урахуванням вимог безпеки та гарантування підтримки високого рівня захищеності планується розглянути в наступних роботах авторів.

ВИСНОВКИ

Впровадження сучасних інформаційних технологій у навчанні дозволяє досягти запланованих результатів тільки за умови надійної, безпечної та продуктивної роботи всієї IT-інфраструктури. До неї пред'являються всі зростаючі вимоги підвищення продуктивності, надійності та захищеності при постійному збільшенні обсягів інформації, що обробляється.

У роботі досліджено проблеми захищеності сучасних LMC систем з урахуванням вразливостей програмного забезпечення, навчального контенту, результатів навчання та персональних даних учасників освітнього процесу, а також надання рекомендацій для організації захисту навчального електронного ресурсу.

Виявлено та проаналізовано проблеми захисту систем LMC з урахуванням сучасних загроз інформаційній безпеці.

Для семи найбільш поширених систем дистанційного навчання й систем керування навчанням виділено й з точки зору інформаційної безпеки систематизовано вразливості, характерні для систем.

Показано, що практично всі системи LMC, які входять в топ-7 найбільш поширених освітніх систем мають можливість встановлення на хмарних платформах. Застосування цих можливостей повинно сприяти підвищенню безпеки навчальних систем, уніфікації процедур захисту, доступності й надійності роботи й, як наслідок, підвищенню якості навчального процесу.

Але LMC системи - це складні структури. Застосування принципів розподілу й відокремлення різних частин систем з їх розміщенням на різних хмарних ресурсах є темою подальших досліджень.

ЛІТЕРАТУРА

[1] M. Averkina, Y. Lykhosherstova, “Digital platforms in interactive learning”, *Modeling the development of the economic systems*, vol. 2023, no. 1, pp. 128-132, 2023.

[2] Yong Chen, Wu He, “Security Risks and Protection in Online Learning: A Survey”, *The International Review of Research in Open and Distance Learning*, vol. 14, no. 5, pp. 108-127, 2013. DOI: 10.19173/irrodl.v14i5.1632

[3] О.О.Будік, В.Ф. Чекурін, “Специфічні загрози інформаційній безпеці систем електронного навчання”, *Вісник Національного університету "Львівська політехніка" Автоматика, вимірювання та керування*, no. 741, pp. 71-76, 2012. [Онлайн]. URL: http://surl.li/rlaos Дата звернення: 10.01.2024 .

[4] Kassid Asmaa, Elkamoun Najib, “E-Learning Systems Risks and their Security”, *International Journal of Computer Science andInformation Security (IJCSIS)*, vol. 14, no. 7, pp. 194-200, 2016.

[5] С.Жовтя, О. І. Полотай, “Програмний захист інформації в системі дистанційного навчання Moodle на основі операційної системи CentOS”, LSULS Digital Repository: Програмний захист інформації в системі дистанційного навчання Moodle на основі операційної системи CentOS (ldubgd.edu.ua). – 2015

[6] N. H. Phuoc Dai, A. Kerti, and Z. Rajnai, “ E-Learning Security Risks and Countermeasures”, *Emerging Research and Solutions in ICT*, vol. 1, no. 1, pp. 17-25, 2020. DOI: 10.20544/ERSICT.01.16.P02.

[7] O.Harasymchuk, I. Opirskyy, Y.Sovyn, I. Tyshyk, Y. Shtefaniuk, “Організація захисту результатів контролю знань в системах дистанційного навчання”, *Кібербезпека: освіта, наука, техніка*, vol. 2, no. 10, pp. 144-157, 2020. Doi: 10.28925/2663-4023.2020.10.144157.

[8] F. Schwarz, “E-Learning in den Ingenieurwissenschaften – Entwicklung”, Anwendung und Evaluation einer internetbasierten Lernumgebung: Doktor Ingenieur, 2009.

[9] Сім найкращих альтернатив Google Classroom, 2024. [Онлайн]. URL: http://surl.li/rlasn/. Дата звернення: 10.01.2024.

[10] Що таке Edpuzzle, 2024. [Онлайн]. URL: http://surl.li/rlatq. Дата звернення: 10.01.2024.

[11] О. Бєлов, М.М. Делембовський, Організація захисту і безпеки в системі «Moodle» Київський національний університет будівництва і архітектури, 2024. [Онлайн]. URL: http://surl.li/rlaum. Дата звернення: 10.01.2024.

ISSN 2786-9024

[12] Інформація з безпеки Moodle.org, URL, 2024. [Онлайн]. URL: http://surl.li/rlavn. Дата звернення: 10.01.2024.

SECURITY AND PROTECTION OF EDUCATIONAL LMC SYSTEMS

Nataliia Maslova, Olena Liubimenko

Information technologies play a significant role in the educational process and ensuring quality learning outcomes in the context of distance education. A modern trend is the development of interactive systems, which include elements of audio and video materials, graphics, presentations, internet links, materials from various sources, and in different formats. The addition of interactive content to educational systems increases the risks of information security in modern LMC systems. Emphasis is placed on the need to ensure the protection of software in systems, educational content hosted within them, learning outcomes, and participants' personal data. When building protection, attention should be paid to potential threats such as unauthorized access, flawed software, poor plugin updates, material copying, fraud, and cyber-attacks. Vulnerabilities of the most widespread modern LMC systems are analyzed. Attention is given to the possibilities of using cloud technologies to host system modules, educational materials, or learning outcomes on cloud services. Cloud application guarantees constant access to the learning system, reliable storage of materials, additional protection of confidential and personal information. Therefore, to ensure effective and safe learning considering modern interactive approaches to presenting educational materials, it is important to develop systems and implement technologies that provide constant access to educational content, reliable data protection (including personal data), information integrity, and compliance with confidentiality principles.

Keywords: *LMC systems, interactive learning, vulnerabilities of distance learning systems, protection, cloud technologies, confidentiality, accessibility, integrity, reliability, information, personal data, educational content.*

REFERENCES

[1] M. Averkina, Y. Lykhosherstova, "Digital platforms in interactive learning", *Modeling the development of the economic systems*, vol. 2023, no. 1, pp. 128-132, 2023.

[2] Yong Chen, Wu He, "Security Risks and Protection in Online Learning: A Survey", *The International Review of Research in Open and Distance Learning*, vol. 14, no. 5, pp. 108-127, 2013. DOI: 10.19173/irrodl.v14i5.1632.

[3] O.O.Budik, V.F. Chekurin, "Spetsyfichni zahrozy informatsiinii bezpetsi system elektronnoho navchannia", Visnyk Natsionalnoho universytetu "Lvivska politekhnika" Avtomatyka, vymiriuvannia ta keruvannia, no. 741, pp. 71-76, 2012. [Online]. URL: http://surl.li/rlaos. Accessed: 10.01.2024. (In Ukranian).

[4] Kassid Asmaa, Elkamoun Najib, "E-Learning Systems Risks and their Security", *International Journal of Computer Science andInformation Security (IJCSIS)*, vol. 14, no. 7, pp. 194-200, 2016.

[5] S.Zhovtia, O. I. Polotai, "Prohramnyi zakhyst informatsii v systemi dystantsiinoho navchannia Moodle na osnovi operatsiinoi systemy CentOS", LSULS Digital Repository: Prohramnyi zakhyst informatsii v systemi dystantsiinoho navchannia Moodle na osnovi operatsiinoi systemy CentOS (ldubgd.edu.ua), 2015. (In Ukranian).

[6] N. H. Phuoc Dai, A. Kerti, and Z. Rajnai, " E-Learning Security Risks and Countermeasures", *Emerging Research and Solutions in ICT*, vol. 1, no. 1, pp. 17-25, 2020. DOI: 10.20544/ERSICT.01.16.P02.

[7] O.Harasymchuk, I. Opirskyy, Y.Sovyn, I. Tyshyk, Y. Shtefaniuk, "Orhanizatsiia zakhystu rezultativ kontroliu znan v systemakh dystantsiinoho navchannia", *Kiberbezpeka: osvita, nauka, tekhnika*, vol. 2, no. 10, pp. 144-157, 2020. Doi: 10.28925/2663-4023.2020.10.144157. (In Ukranian).

[8] F. Schwarz, "E-Learning in den Ingenieurwissenschaften – Entwicklung", Anwendung und Evaluation einer internetbasierten Lernumgebung: Doktor Ingenieur, 2009.

[9] Сім найкращих альтернатив Google Classroom, 2024. [Online]. URL: http://surl.li/rlasn/. Accessed: 10.01.2024.

[10] Shcho take Edpuzzle, 2024. [Online]. URL: http://surl.li/rlatq. Accessed: 10.01.2024. (In Ukranian).

[11] O. Bielov, M.M. Delembovskyi, Orhanizatsiia zakhystu i bezpeky v systemi «Moodle» Kyivskyi natsionalnyi universytet budivnytstva i arkhitektury, 2024. [Online]. URL: http://surl.li/rlaum. Accessed: 10.01.2024. (In Ukranian).

[12] Informatsiia z bezpeky Moodle.org, URL, 2024. [Online]. URL: http://surl.li/rlavn. Accessed: 10.01.2024. (In Ukranian).

 https://doi.org/10.5281/zenodo.10814232

BUSINESS MODELS IN STARTUP GROWTH FORECASTING

Yakiv Baytelman[1]

[1] Department of Applied Mathematics and Informatics, Donetsk National Technical University, Lutsk, Ukraine
E-mail: yakiv.baitelman.kita@donntu.edu.ua

Отримано 31.12.2023
Прийнято до публікації 29.01.2024
Опубліковано 01.04.2024

ABSTRACT

A start-up is the IT industry specific form of commercial enterprise with its own peculiarities not observed in other forms of business ventures. Hands-on purposes require a toolset for start-up growth modelling and forecasting as well as for their evaluation; business models belong to such tools. Advantages of business modelling is not always obvious to start-up founders, up to the moment when it becomes a part of requirements from investors. A form that allows quick modelling greatly simplifies evaluation and triage of start-ups by investors and thus can become one of formatting factors for regional or national start-up ecosystems in Ukraine. Based on the recent publications this research offers an example of a business model and suggests several key aspects for more practical application of business modelling for the sake of start-up growth forecasting, including expression of main milestones in the financial projection.

Keywords*: start-up, innovations, business model, business plan, investment, ecosystem.*

INTRODUCTION

The term "start-up" is usually used in the context of innovations in software development, sometimes also including a certain amount of hardware production, and is applied to a newly established venture in its early stages. Start-ups focus on offering a new solution to an existing problem or on creating a totally new demand. From the customer perspective they address either individual customers (B2C) or other businesses (B2B) or individual customers of another service provider (B2B2C). Although purely software start-ups call their solutions "products" and this term gained wide adoption in the industry those products are services while hardware solutions are closer to the classic understanding of a product. The inventor of the word "start-up" is unknown, the very first track of usage of this word dates to the 1970s. It was popularised by various conferences and other events to finally become a widely recognizable lexeme in different languages. The main difference of a start-up from a "traditional" business venture finds its grounds in the innovativeness of the offered solution that is why uncertainty is the key differentiator with every stage requiring validation, from the feasibility studies to creation of a minimal viable product (MVP), its route to market and further upscaling. If in "traditional" business the question covers "know how" someone else has already done it and the aim of the enterprise is to reproduce the same results in a new location or with higher efficiency, start-ups must deal with discovery of implementation possibility and costs. The said puts start-ups into the high-risk enterprise category, but since the early days of "garage" start-ups the high-tech industry has developed a complex of approaches, methods and instruments for start-up growth which collectively can be addressed as an ecosystem.

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10814232

In the pre-start-up epoch inventions appeared as the fruit of scientific work under the umbrella of research centres of universities or big industrial corporations due to the high cost of computation equipment, limited access to it and the “analogue” nature of the devices we used. In the early years of post-industrial economy, with the Internet becoming more ubiquitous and personal computers affordable, development and route to market of purely software solutions could no longer be a privilege of the few research centres, so we witnessed emergence of unimaginable earlier computer programs created by small companies (early 2000s). Some of those companies and programs soon became forgotten. Others revolutionised the industry, for example, Skype, a small Estonian company back then, created a solution for free Internet calls, later this company was acquired by Microsoft and once a profitable market of expensive international phone calls got almost fully replaced with a new demand for countless solutions for free voice and video calls. Then computers decreased in size transforming into early smartphones opening a new opportunity for more start-ups with more innovative ideas (2000 – 2010). After that the main manufacturers of mobile phones introduced so-called application markets for their platforms effectively forcing software developers to use these markets as the only possible distribution options for their solutions. Over time Google’s and Apple’s restrictions to access programmatically device components and sensors (e.g. location services) became quite strict. Today software developers are constrained with highly regulated legal implementation options and tightly defined technical features of the low level (operating system) capabilities; thus, innovations are once again in the history of humanity streamlined – if not controlled – by the industry giants and corporations. In this research the author elaborates on some pragmatic aspects of such limitations from the economic perspective.

ANALYSIS OF RECENT RESEARCH AND PUBLICATIONS, PROBLEM STATEMENT

The term “business model” is widely used in scientific and popular business literature and is commonly understood as a theoretical construct that represents a specific way of customer acquisition and revenue generation [1, 2]. In practical application a business model is required by investors and expected as a part of the pitch deck, ideally in the form of an excel spreadsheet with formulas allowing to modify input parameters and see the output results. This is where a shift from “static” business plan documents to more dynamic modelling is clearly observed. Business models should help to understand the revenue generation plan, be aligned with well-defined short term-objectives and strategic long-term goals and the company's mission, at the same time remaining flexible to adapt to changes [3].

The industry knows a few cases of very successful internet companies which achieved significant growth and global adoption without any precise economic plan, the most famous one is Twitter. The idea of the founders was as “simple” as accumulating a huge number of users (not to be confused with customers!) and figuring out monetization later. Eventually Twitter's leadership team decided to sell advertisements and offer premium services to their users on a subscription basis [4].

In the light of the customer lifecycle the key metrics include the customer acquisition cost and the lifetime value of a customer [5]. On the other side of the scale unit economics shall be estimated assuming the worst possible parameters and including all liabilities. As the result of an accurate reflection of the above-mentioned factors a start-up business model shall provide a clear view on the profitability and allow to run simulations and thus enable evaluation firstly for the founders and then for potential investors. A frequent mistake of technical founders lies in their failure to assess and evaluate their idea from the economic perspective, so their innovative and technically doable project makes no sense moneywise [6].

Lean start-up is believed to be one of the most efficient growth methodology, it uses a different compared to traditional businesses approach based on assumptions or vision of the founders and rapid testing of such assumptions. “While existing firms execute a business model, start-ups look for one” – this translates into a series of small iterations in development of different features of a software solution, each of them has to be promptly validated through MVP tests involving real users. Assumptions are validated, the use of scarce resources is optimised and building a product that no one is ready to pay for is avoided. The other problem is related to the lack of commonly acceptable definition of success. While investors basically expect to see some level of market traction, start-up and investor communities all over the world do not share any precise understanding of what would suffice as such market traction to clearly categorise ventures according to risks vs. profitability; the lack of any unification does not help to scale the process of triaging start-ups for investment purposes raising the same challenge of evaluating each new start-up [7].

In recent years the gap between business model theory and its hands-on usage by entrepreneurs has been closing: founders tend to convert their abstract vision into pragmatic rules guided by business models to help them in making more accurate decisions under uncertainty, in task prioritising and resource management [8, 9]. A true breakthrough is owed to a wide range of tools for automation of user data collection

 https://doi.org/10.5281/zenodo.10814232

and analysis often referred to as data driven business model development, once affordable for big corporations but now open to a broader audience. Moreover, for a start-up to fully benefit from user data analysis and corresponding business insights the process of data collection should be planned for and implemented at early stages [10]. Numerous business intelligence tools are available to entrepreneurs and scholars for free or for a quite affordable fee so working with user data has become a comparatively easy exercise leaving no excuses to omit it from start-up development routines [11, 12, 13]. Despite a strong temptation to utilise Artificial Intelligence in business modelling we should not rely blindly on any answers or conclusions provided by AI but rather consider them as probability-based predictions in need of validation [14].

Attempting to create innovative solutions, digital entrepreneurs often count upon such resources as open data or free services although access to them is not guaranteed and can become unavailable at any moment [15]. This should not be skipped from the start-up data model because growth or even existence of the start-up depends greatly on 3rd-parties whose terms of service can change unexpectedly [16, 17].

Given noteworthy rise of blockchain start-ups, the author searched for mentors' guidance on business modelling for companies in this industry supported with some feedback from investors, below are the key focus items well valid for ventures in other industries:

1. Identifying revenue streams such as transaction or subscription fees, advertising revenue, referral schemes.
2. Estimating potential income.
3. Categorising costs as fixed or variable.
4. Calculating net income through subtracting expenses from revenue.
5. Analysing the burn rate to check financial health and know the point in time when the business runs out of money.
6. Evaluating the start-up's runway as the indication of time left to achieve profitability supposing no new funding is available.
7. 3-year financial projections including assumptions about revenue and expense growth.
8. Improving accuracy of the business model through use of realistic assumptions backed by metrics.

Common errors include misunderstanding the technology, neglecting regulatory requirements, expecting too high user adoption too quickly [18, 19].

The problem: on the one hand modern start-ups face various challenges in their attempts to evaluate the time, resources and costs needed to create and take to the market something previously unknown and establish a steady demand for it. On the other hand, they benefit from the existing ecosystem but at the same time remain somewhat constrained. Start-ups work against the clock trying to accomplish their goals before they run out of funds in highly volatile environments. Often access to funding is a bigger problem than proving solution feasibility. Sometimes innovative products or services belong to an area where legislation still has to catch up with the technology, (e.g. cryptocurrencies). Areas like defence or healthcare make field tests hard. Global nature of the Internet provides an opportunity to address potential customers regardless of their physical location or nationality, albeit this gain comes with the pain of imposing the same requirements on industry giants and tiny new companies, e.g. data protection regulations. No matter what solution a start-up works at, the competition would be huge, to say the least. This is a short scope of hurdles and risk factors start-ups must deal with, each of them is worth studying in the light of possibility to model and forecast a start-up growth for deeper understanding in general and for defining those specific issues for new Ukrainian tech companies in particular so that upcoming theoretical work on Start-up Ecosystem in Ukraine would find further elaboration. Standard approaches widely used for "traditional" businesses would not suffice for start-ups mostly because of numerous unknown conditions start-ups have to research in the process of their growth.

The aim of this research is to drill down the specifics of start-up development based on documented case studies and best practices articulated by investors and mentors, outline the key approaches for business modelling in start-ups and suggest a simplified example of a business model to illustrate how it can be used by entrepreneurs for self-control and planning, by investors for start-up evaluation and how it could become a tool for general adoption inside regional and national start-up ecosystem.

MATERIALS AND RESEARCH METHODS

Methods of this work included structural and comparative analysis and elements of computer modelling. Recent publications in scientific literature and on specialised online resources were studied with informational and analytical approaches.

RESULTS OF THE RESEARCH

In the past years research around the globe have been strongly advocating the advantages of business modelling in young and growing tech companies however a group of questions remain unclear. To begin with, let us bring into the spotlight a common habit to identify users of a software solution as customers of the company providing this solution. Obviously, users are the ones who interact with the service, create and consume content, generate traffic, leave certain

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10814232

digital footprints in the system, from simple browsing history to complex unique behaviouristic patterns. Users should be understood as a liability of the company because the business incurs costs of user acquisition and maintenance, for instance a paid ad to find a new user, cost of storing the user's content, cost of traffic the user generates, computation power needed to process the user's data, etc. Less obvious but nevertheless critically important for building an adequate business model is the transformation of users into customers, which happens only once they start paying for the service. Since users do not pay, the company must onboard other customers who will. In a social network, a user accesses the service for free and thus is not a customer, a paying advertiser is. Each of us is a user of free Gmail, Facebook, Twitter, or LinkedIn but only some of us are customers, those who pay for Google for Workspace, ads in Facebook, Twitter Blue or LinkedIn Premium. So, it is not enough for a start-up to design a business model based on growth of its free service user base, the plan to convert free users into paying customers should be in place which sets forth probably the main question for start-up surviving: what is the user to customer conversion rate? Business modelling together with lean start-up methodology can help in finding the answer otherwise entrepreneurs get stuck in guesstimating. Quick experiments to validate user adoption of each new small (MVP level) feature are to minimise risks of building something nobody would want; simulations of financial projections provide a helicopter view on the entire business operations.

Let us consider a somewhat simplified high-level vision of a start-up:

1. The MVP will be created in a 3-months term, it will offer sufficient functionality to start onboarding of both freemium and paying customers. Further research and development work will continue to enrich the solution with new features.
2. 3 developers are needed full time (back-end, Android, iOS), monthly salary for each of them is USD 3,000, it is verified, contracts with these developers are negotiated and secured.
3. Purchase of 3 PCs for the developers (3 * USD 1,500 = USD 4,500), the prices are verified.
4. Acquisition of freemium users and paying customers shall be outsourced to an agency who requests a fixed price per each registered user and offers theoretically unlimited number of them.
5. There will be some legal and accounting costs, the amounts are verified.
6. Monthly cost of user maintenance can be measured experimentally and calculated per user, so it can be considered as known.
7. Monthly customer fee shall be somewhat between USD 10 and 20. A higher rate will not compete with similar services provided by competitors; a smaller will make no sense.
8. The business must become profitable within its first year.

The number of both freemium users and paying customers is an open question, so is the aggressiveness of their onboarding. The amount of the monthly customer fee is not chosen. Without knowing the answers to these questions, it is impossible to prognose the spendings and earnings, which in its turn prevents from understanding how much money this start-up plans to raise. Obviously, due to the big number of unknowns, only multiple simulations can provide enough grounds for decisions.

First, unit economics should be understood and modelled (Table 1).

Table 1. Unit Economics

Parameter	One time	Daily	Monthly
A new freemium user acquisition, USD	2		
A new paying customer acquisition, USD	5		
Hosting and traffic per user, USD			0.075
Customer fee, USD			19.9
Engineer cost, USD		150	
Legal cost, USD			200
Accounting cost, USD			100

As it was mentioned above, the cost of customer acquisition, the cost of user hosting and traffic, the cost of accounting, legal and engineering resources, and customer free can be considered as known and presented as one-time fee or recurring monthly or daily fee depending on their nature. Each of these parameters in the model can be changed for the purpose of simulation of the financial projection. For the sake of simplicity in this work only the customer fee will be modified.

In the first simulated projection (Fig. 1) the customer fee is set to be expensive, USD 19.9, the process of freemium user acquisition is aggressive while the onboarding of paying customers is slow. The total number of users is as high as 75 thousand. In October the monthly sales outgrow the monthly spendings, which means that until that moment the accumulated spendings should be covered by the investment, hence the amount this start-up needs to raise is USD 181K. The business becomes profitable in December with the total budget of USD 310K. Although the annual net

 https://doi.org/10.5281/zenodo.10814232

income is minimal, the last month shows the income equals USD 76K, assuming nothing changes, the business will generate this amount each subsequent month.

Metrics	*Jan*	*Feb*	*Mar*	*Apr*	*May*	*Jun*	*Jul*	*Aug*	*Sep*	*Oct*	*Nov*	*Dec*	*Total*
Engineering mandays {input}	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	
Enginering cost, USD	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	
Legal and accounting, USD	300	300	300	300	300	300	300	300	300	300	300	300	
Device purchase, USD	4,500	0	0	0	0	0	0	0	0				
Increase of freemium users {input}	**0**	**0**	**0**	**1,000**	**3,000**	**5,000**	**10,000**	**10,000**	**10,000**	**10,000**	**10,000**	**10,000**	
Increase of paying customers {input}	**0**	**0**	**0**	**0**	**100**	**100**	**100**	**300**	**1,000**	**1,000**	**1,500**	**2,000**	
Cost of freemium user acquisitions, USD	0	0	0	2,000	6,000	10,000	20,000	20,000	20,000	20,000	20,000	20,000	
Cost of paying customer acquisition, USD	0	0	0	0	500	500	500	1,500	5,000	5,000	7,500	10,000	
Total paying customers	0	0	0	0	100	200	300	600	1,600	2,600	4,100	6,100	
Customer fee, USD	0	0	0	0	1,990	3,980	5,970	11,940	31,840	51,740	81,590	121,390	
Total users (freemeum + paying)	0	0	0	1,000	4,100	9,200	19,300	29,600	40,600	51,600	63,100	75,100	
Hosting and traffic, USD	0	0	0	75	307.5	690	1,448	2,220	3,045	3,870	4,733	5,633	
Monthly spendings, USD	13,800	9,300	9,300	11,375	15,608	19,990	31,248	33,020	37,345	38,170	41,533	44,933	305,620
Monthly sales, USD	0	0	0	0	1,990	3,980	5,970	11,940	31,840	51,740	81,590	121,390	310,440
Spendings so far, USD	13,800	23,100	32,400	43,775	59,383	79,373	110,620	143,640	180,985	219,155	260,688	305,620	
Sales so far, USD	0	0	0	0	1,990	5,970	11,940	23,880	55,720	107,460	189,050	310,440	

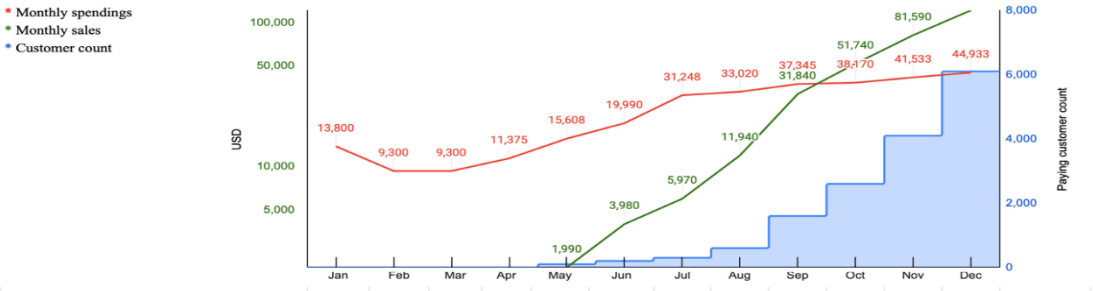


Fig. 1. An expensive fee, slow customers onboarding

The second simulated projection (Fig. 2) differs from the first one in a reduced customer fee, a smaller total user count, but most importantly, in a gradual reducing of the engineering resources. Similarly to the previous scenario, the profitability is achieved. The amount to raise is USD 136K, the annual budget is 248K, the income to be generated in December and each subsequent month is USD 59K.

Both scenarios were designed with the plan in mind to establish a big free user base with the intention to convert in the future free users into paying customer, which is an assumption that needs validation. A plan to reduce engineering efforts can be rather risky and will most likely be rejected by investors because keeping engineering effort on the level of 20 man-days per month does not address situations when this single engineer can become unavailable. Also planning for team reduction is not aligned with future expansion and upscaling. Another observation from these 2 scenarios refers to the budget for the whole project in the first year, raising a question of possibility to achieve project profitability with smaller investments.

ISSN 2786-9024

Metrics	*Jan*	*Feb*	*Mar*	*Apr*	*May*	*Jun*	*Jul*	*Aug*	*Sep*	*Oct*	*Nov*	*Dec*	*Total*
Engineering mandays {input}	**60**	**60**	**60**	**60**	**50**	**40**	**20**	**20**	**20**	**20**	**20**	**20**	
Enginering cost, USD	9,000	9,000	9,000	9,000	7,500	6,000	3,000	3,000	3,000	3,000	3,000	3,000	
Legal and accounting, USD	300	300	300	300	300	300	300	300	300	300	300	300	
Device purchase, USD	4,500	0	0	0	0	0	0	0	0				
Increase of freemium users {input}	**0**	**0**	**0**	**1,000**	**1,000**	**2,000**	**5,000**	**10,000**	**10,000**	**10,000**	**10,000**	**10,000**	
Increase of paying customers {input}	**0**	**0**	**0**	**0**	**100**	**100**	**100**	**300**	**1,000**	**1,000**	**1,500**	**2,000**	
Cost of freemium user acquisitions, USD	0	0	0	2,000	2,000	4,000	10,000	20,000	20,000	20,000	20,000	20,000	
Cost of paying customer acquisition, USD	0	0	0	0	500	500	500	1,500	5,000	5,000	7,500	10,000	
Total paying customers	0	0	0	0	100	200	300	600	1,600	2,600	4,100	6,100	
Customer fee, USD	0	0	0	0	1,590	3,180	4,770	9,540	25,440	41,340	65,190	96,990	
Total users (freemeum + paying)	0	0	0	1,000	2,100	4,200	9,300	19,600	30,600	41,600	53,100	65,100	
Hosting and traffic, USD	0	0	0	75	157.5	315	698	1,470	2,295	3,120	3,983	4,883	
Monthly spendings, USD	13,800	9,300	9,300	11,375	9,958	10,615	14,498	26,270	30,595	31,420	34,783	38,183	240,095
Monthly sales, USD	0	0	0	0	1,590	3,180	4,770	9,540	25,440	41,340	65,190	96,990	248,040
Spendings so far, USD	13,800	23,100	32,400	43,775	53,733	64,348	78,845	105,115	135,710	167,130	201,913	240,095	
Sales so far, USD	0	0	0	0	1,590	4,770	9,540	19,080	44,520	85,860	151,050	248,040	

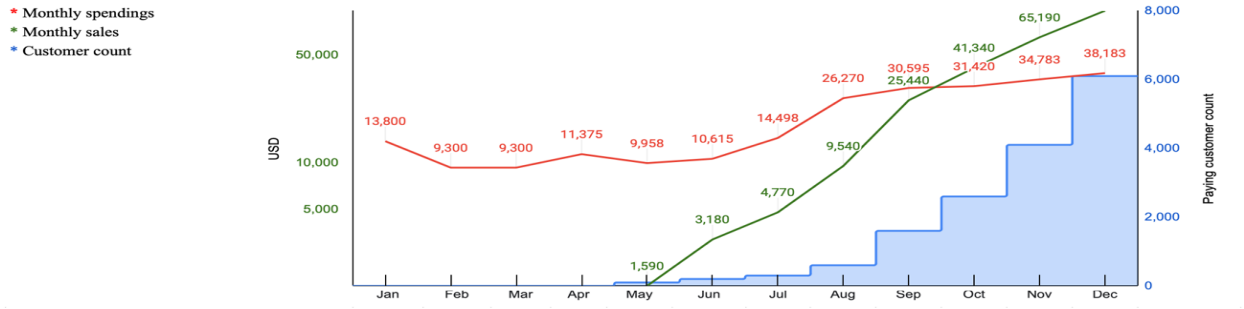


Fig. 2. Reducing engineering resources, slow onboarding

Fig. 3 shows a projection with a considerably lower budget achievable though skipping freemium user acquisition and focussing on paying customers only. In this case the start-up seeks to raise only USD 71 K.

Metrics	*Jan*	*Feb*	*Mar*	*Apr*	*May*	*Jun*	*Jul*	*Aug*	*Sep*	*Oct*	*Nov*	*Dec*	*Total*
Engineering mandays {input}	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	**60**	
Enginering cost, USD	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	9,000	
Legal and accounting, USD	300	300	300	300	300	300	300	300	300	300	300	300	
Device purchase, USD	4,500	0	0	0	0	0	0	0	0				
Increase of freemium users {input}	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	
Increase of paying customers {input}	**0**	**0**	**0**	**0**	**100**	**100**	**300**	**500**	**500**	**500**	**500**	**500**	
Cost of freemium user acquisitions, USD	0	0	0	0	0	0	0	0	0	0	0	0	
Cost of paying customer acquisition, USD	0	0	0	0	500	500	1,500	2,500	2,500	2,500	2,500	2,500	
Total paying customers	0	0	0	0	100	200	500	1,000	1,500	2,000	2,500	3,000	
Customer fee, USD	0	0	0	0	1,590	3,180	7,950	15,900	23,850	31,800	39,750	47,700	
Total users (freemeum + paying)	0	0	0	0	100	200	500	1,000	1,500	2,000	2,500	3,000	
Hosting and traffic, USD	0	0	0	0	7.5	15	38	75	113	150	188	225	
Monthly spendings, USD	13,800	9,300	9,300	9,300	9,308	9,315	10,838	11,875	11,913	11,950	11,988	12,025	130,910
Monthly sales, USD	0	0	0	0	1,590	3,180	7,950	15,900	23,850	31,800	39,750	47,700	171,720
Spendings so far, USD	13,800	23,100	32,400	41,700	51,008	60,323	71,160	83,035	94,948	106,898	118,885	130,910	
Sales so far, USD	0	0	0	0	1,590	4,770	12,720	28,620	52,470	84,270	124,020	171,720	

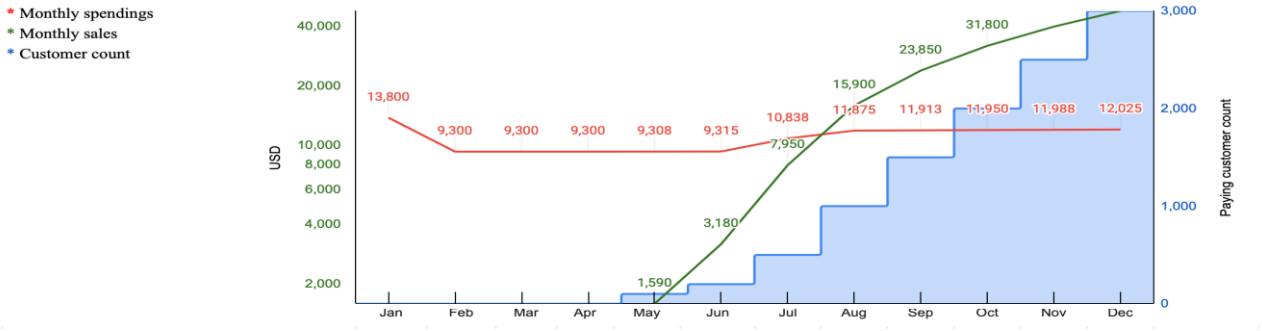


Fig. 3. A lower budget, no freemium users

Experimenting with variable values of the input parameters of this simplified model gives us unlimited number of scenarios, more or less aggressive, cheap and expensive, with different levels and types of risks.

Now let us bring the above business model closer to reality and introduce an extra layer of complexity associated with processing of customer payments. From the perspective of payment frequency, economics of a start-up can be subscription or transaction based. Surely, any business would prefer to bind their customers with long-term obligations in the form of automatically renewed subscriptions which makes sense in content access solutions (e.g. Netflix).

ISSN 2786-9024

 https://doi.org/10.5281/zenodo.10814232

Transaction fees work better with solutions where the number of said transactions per customer is unpredictable, the commission (transaction fee) depends on the cost of transaction and where such approach is traditionally justified (e.g. crypto payments). Regarding customers' money access a start-up has the following choice of options and each of them is characterised by the value of commission, cost of implementation and maintenance, options to finally withdraw the funds to the start-up's account:

- Old-fashioned wire transfer. It is applicable in B2B and not very popular. The only advantage is that the money is directly deposited to the company's bank account.
- Payment card processing. Heavily regulated thus start-ups prefer to work via a 3rd party who specialises in processing card payments (e.g. Stripe, PayPal), and only industry giants implement their own direct processing of payments.
- Local payment solutions (in Ukraine Privat24, in Kenya Mpesa and so on). The downside of this method is low scalability. A solution for Ukraine will not work in Germany and for a start-up to go global this needs to be redone.
- Payment through the application market (Google Play, App Store), the one and only legal solution for many mobile applications, especially when the payment is for a virtual service. This is one of the earlier mentioned constraints both Apple and Google put on the app developers.
- Payments in crypto currencies. Problematic from the legislative standpoint and depends on jurisdiction.
- Cash payments shall be mentioned as a very poor option existing only due to some local regulations (e.g. an option to pay cash to an Uber driver).

Our model says nothing about the location of the customers, so we assume they can be anywhere in the world therefore we need a universal payment solution. We know that our service is implemented in the form of a mobile application (the plan mentions Android and iOS developer) and correspondingly in-app purchases shall be processed by the billing systems of Google Play and App Store. In general, each of them takes 30% commission with the exceptions below:

1. Both Apple and Google offer a discounted fee of 15% if the business makes less than USD 1 million in annual net app revenue.
2. These revenue share rules only apply to digital goods and services excluding:

a. physical goods such as groceries, clothing, houseware, or electronics;

b. physical services such as bus or train tickets, gym memberships, food delivery, cinema tickets, hotel booking;

c. payment of a credit card or utility bill.

Another risk factor should be included into the business model though it is not always clear how to quantify it: reliance on 3rd-party services (APIs). For example if the start-up product offers navigation from point A to point B as its key feature meaning that without it the product does not exist, from the engineering perspective this can be achieved with Google Maps as well as Apple MapKit and also some other data providers (OpenStreetMap, etc), the fee varies and usually depends on the volume and chosen technology, in any case the cost can be calculated based on predicted usage and the choice of provider. If the product is actually nothing more but an add-on on top of a 3rd-party service, for instance, a solution for certain automation in calculating YouTube video ratings and showing only the relevant videos to the users based on their interests and location, apparently, this functionality fully depends on YouTube API. If at any moment in time YouTube decides to modify their API additional work shall be required to keep up with the changes, if YouTube stops serving through their APIs such critical for this solution information as video ratings the whole solution becomes absolutely blocked, no remedy is possible even theoretically because the entire business is built on one only irreplaceable 3rd-party component. Such risks are often underestimated by start-up founders while investors identify them easily through preliminary due diligence questionnaires. Such risk factors should be treated as "0 multipliers" in the business model as they can override all and any other terms.

Regarding the planning horizon: for illustration purposes the suggested above business model covers 12 months only. In reality securing a seed funding round can take a few months with following rounds requiring much more time from the moment of request submitting to receiving money into the bank account. With this in mind, a proper business planning and modelling should be done for 3 years with these milestones indicated in the financial projection:

1. Completion of MVP.
2. Start and end of each round of customer acquisition.
3. Each significant product release.
4. Monthly sales reach the level of monthly spendings (profitability).
5. Net profit reaches the level of the total cost of investment (ROI).

From the ecosystem perspective, business modelling – particularly if designed and structured according to common principles and requirements for a given region or country – can become a powerful tool for both entrepreneurs and investors, serving the role of shaping start-ups as qualified for funding and capable of rapid growth. Such common

ISSN 2786-9024

requirements are not to be expected to emerge from legislation or regulations but rather become a result of educational work provided via regional and national start-up hubs and backed by initiatives for support of innovative entrepreneurship on the national level. In this educational effort a significant place naturally belongs to universities and colleges which can contribute their strong scientific potential, whose today's students will form tomorrow's community of new entrepreneurs and workforce.

CONCLUSIONS

Spreading awareness of the advantages of business modelling alongside pragmatic advice and guidance for start-up founders is needed in order to increase effectiveness of start-ups. This activity can be offered through joint efforts of academic institutions and start-up hubs under regional and national initiatives for support of innovative businesses and can contribute to development of start-up ecosystems.

REFERENCES

[1] Faster Capital. Creating an Economic Model for Your Startup, 2023. [Online]. URL: http://surl.li/rncpm. Accessed: 15.12.2023.
[2] M.Kravchenko, V.Sydorchuk, "Practical approaches to business modeling of innovation projects", *Economic Scope*, 160, 65-72, 2020. [Online]. URL: http://surl.li/rncvi. Accessed: 15.12.2023. (In Ukranian).
[3] Faster Capital. Create a business model that will win over investors, 2023. [Online]. URL: http://surl.li/rncwg. Accessed: 15.12.2023.
[4] D.Pereira. Twitter Business Model. The Business Model Analyst, 2023. [Online]. URL: http://surl.li/rncxx. Accessed: 15.12.2023.
[5] B.Waters. Create an Economic Model, 2021. [Online]. URL: http://surl.li/rndcz. Accessed: 15.12.2023.
[6] T.Sak, N.Hrytsiuk, "Start-up business model: essence, types and opportunities for application", *Scientific Bulletin of Polissia*, 1(24), 93-107, 2022. [Online]. URL: http://surl.li/rndex. Accessed: 15.12.2023. (In Ukranian).
[7] W.Kenton. Lean Startup: Defined, How It Differs From a Traditional Business, 2022. [Online]. URL: http://surl.li/rndhg. Accessed: 15.12.2023.
[8] A.Ghezzi, A.Cavallaro, A.Rangone, R.Balocco, "A Comparative Study on the Impact of Business Model Design & Lean Startup Approach versus Traditional Business Plan on Mobile Startups Performance", *ICEIS 2015 - 17th International Conference on Enterprise Information Systems*, vol. 3, 196-203, 2015. [Online]. URL: http://surl.li/rndlb. Accessed: 15.12.2023.
[9] A. Ghezzi, How Entrepreneurs make sense of Lean Startup Approaches. Business Models as cognitive lenses to generate Fast and Frugal Heuristics. *Technological Forecasting and Social Change*, vol. 161, 2020. [Online]. URL: http://surl.li/rndok. Accessed: 15.12.2023.
[10] B.Marcinkowski, B.Gawin, "Data-driven business model development – insights from the facility management industry". *Journal of Facilities Management*, vol. 19 No. 2, pp. 129-149, 2021. [Online]. URL: http://surl.li/rndrz. Accessed: 15.12.2023.
[11] Google. Introduction to analysis and business intelligence tools. [Online]. URL: http://surl.li/rndtv. Accessed: 15.12.2023.
[12] Microsoft. What to ask about business intelligence tools. [Online]. URL: http://surl.li/rndvr. Accessed: 15.12.2023.
[13] Amazon QuickSight. [Online]. URL: http://surl.li/rndww. Accessed: 15.12.2023.
[14] D. Sjödin, V.Parida, M.Palmié, J.Wincent, "How AI capabilities enable business model innovation: Scaling AI through co-evolutionary processes and feedback loops", *Journal of Business Research*, vol. 134, 574-587. 2021. [Online]. URL: http://surl.li/rneac. Accessed: 15.12.2023.
[15] M.Kamariotou, F.Kitsios, "Bringing Digital Innovation Strategies and Entrepreneurship: The Business Model Canvas in Open Data Ecosystem and Startups", *Future Internet*, vol.14(5), 127. 2022. [Online]. URL: http://surl.li/rneby. Accessed: 15.12.2023.
[16] G.Harrison. The API economy vs. the forces of chaos, 2021. [Online]. URL: http://surl.li/rnefw. Accessed: 15.12.2023.
[17] A.Balaganski. The Dark Side of the API Economy, 2019. [Online]. URL: http://surl.li/rnelo. Accessed: 15.12.2023.
[18] Faster Capital. How To Create A Financial Model For A Blockchain startup, 2023. [Online]. URL: http://surl.li/rneni. Accessed: 15.12.2023.
[19] D.Marikyan, S.Papagiannidis, O.F.Rana, R.Ranjan, "Blockchain: A business model innovation analysis", *Digital Business*, vol. 2, iss. 2, 2022. [Online]. URL: http://surl.li/rneqm. Accessed: 15.12.2023.

БІЗНЕС МОДЕЛІ В ПРОГНОЗУВАННІ РОЗВИТКУ СТАРТАПІВ

Байтельман Я. Л.

Стартап є специфічною для IT-галузі формою функціонування комерційного підприємства із певними власними особливостями, не притаманними іншим формам бізнесів. Практичні цілі диктують потреби в інструментарії для моделювання і прогнозування розвитку, а також оцінювання потенціалу стартапів; бізнес моделі належать до таких інструментів. Переваги бізнес моделювання не завжди є очевидними для засновників стартапів, аж доки бізнес модель не стає частиною вимог з боку інвесторів. Формат, що дозволяє швидке моделювання, суттєво полегшує оцінювання і відбір стартапів інвесторами і таким чином може бути одним з чинників формування регіональної та національної стартап-екосистеми в Україні. На основі новітніх публікацій пропонується приклад бізнес моделі і наводиться ряд ключових аспектів прикладного застосування бізнес моделювання з метою прогнозування розвитку стартапу, включаючи відображення основних вік

 https://doi.org/10.5281/zenodo.10814232

розвитку та факторів ризику у формі фінансової проекції.

Ключові слова: *стартап, інновації, бізнес модель, бізнес план, інвестування, екосистема, фінансова проекція.*

ISSN 2786-9024