УДК 004.05
DOI https://doi.org/10.32689/maup.it.2024.4.9

Максим ДЬЯЧЕНКО
аспірант, Державний торговельно-економічний університет, m.diachenko@knute.edu.ua
ORCID: *0009-0002-0279-2497*

Андрій РОСКЛАДКА
доктор економічних наук, професор,
завідувач кафедри цифрової економіки та системного аналізу,
Державний торговельно-економічний університет, a.roskladka@knute.edu.ua
ORCID: *0000-0002-1297-377X*

ВПРОВАДЖЕННЯ ШТУЧНОГО ІНТЕЛЕКТУ В ПРОЦЕС МЕНЕДЖМЕНТУ ІНЦИДЕНТІВ

Анотація. *Швидкий розвиток технологій вимагає трансформації практик управління IT послугами (ITSM). У цій статті досліджується інтеграція штучного інтелекту для IT Операцій (AIOps) та інтелектуальної автоматизації в рамках ITSM, зокрема, з акцентом на їхній вплив на управління інцидентами, операційну ефективність та загальну якість обслуговування. На основі огляду останніх літературних джерел та кейс-стаді, стаття має на меті надати інсайти щодо переваг, викликів та майбутніх напрямків цих технологій у покращенні IT операцій. Результати дослідження вказують на значний потенціал AIOps та інтелектуальної автоматизації у підвищенні ефективності управління IT послугами, проте реалізація цих технологій вимагає ретельного планування та врахування особливих аспектів. Запропонована методологія впровадження може бути широко використана організаціями для подальшого розвитку напрямку AIOps.*

Метою дослідження *є створення методології інтеграції AI автоматизацій у технічні та бізнес процеси організації. Завдання, які необхідно виконати для цього, включають дослідження існуючих підходів у класифікації і використанні AI автоматизацій у сфері, аналіз існуючих систем і досвід їх впровадження, та опис методології впровадження AI автоматизацій.*

Методологія *включає аналіз літературних джерел та аналіз існуючих варіантів застосування автоматизацій у процесах. Засобами системного аналізу розроблено методологію впровадження таких автоматизацій у бізнес процес.*

Наукова новизна *полягає в адаптації новітніх підходів у менеджменті інцидентів до поточних процесів технологічних організацій.*

Висновки. *В цій роботі запропонована покрокова методологія впровадження в бізнес-процеси автоматизацій на базі штучного інтелекту, розгортаючи інфраструктуру AIOps. Застосовування розробленої методології і проведення подальших кейс-стаді із визначеннях можливих особливостей процесів організацій є перспективним напрямком подальших досліджень.*

Ключові слова: *AIOps, менеджмент інцидентів, штучний інтелект, ITSM, впровадження процесу.*

Maksym DIACHENKO, Andrii ROSKLADKA. IMPLEMENTATION OF ARTIFICIAL INTELLIGENCE INTO THE INCIDENT MANAGEMENT PROCESS

Abstract. *The rapid advancement of technology necessitates the transformation of IT Service Management (ITSM) practices. This article examines the integration of Artificial Intelligence for IT Operations (AIOps) and intelligent automation within ITSM, focusing on their impact on incident management, operational efficiency, and overall service quality. Based on a review of recent literature and case studies, the article aims to provide insights into the benefits, challenges, and future directions of these technologies in improving IT operations. The findings highlight the significant potential of AIOps and intelligent automation to enhance IT service management efficiency; however, their implementation requires careful planning and consideration of specific factors. The proposed implementation methodology can be widely utilized by organizations to further advance AIOps adoption.*

The goal *of the study is to develop a methodology for integrating AI-driven automations into an organization's technical and business processes. The tasks required to achieve this include researching existing approaches to the classification and application of AI automations in the field, analyzing current systems and implementation experiences, and describing the methodology for integrating AI automations.*

The methodology *includes a review of literature and an analysis of existing use cases for automation in processes. Using system analysis tools, a methodology for integrating such automations into business processes has been developed.*

The scientific novelty *lies in the adaptation of cutting-edge approaches in incident management to the current processes of technology-driven organizations.*

Conclusions. *This study proposes a step-by-step methodology for implementing AI-based automation in business processes through the deployment of AIOps infrastructure. Applying the developed methodology and conducting further case studies to identify potential organizational process nuances represent promising directions for future research.*

Key words: *AIOps, incident management, Artificial Intelligence, ITSM, process implementation.*

Вступ. Сучасні IT-інфраструктури стають дедалі більшими та складнішими через постійний розвиток технологій та зміну робочих методів. Підтримка ефективності та надійності в таких середовищах є складним завданням. Багато організацій переходять від традиційних продуктів до надання послуг,

обираючи динамічні комбінації локальних, керованих, приватних та публічних хмарних середовищ. Через такі фактори, як мобільність пристроїв, зміни в середовищах виконання, часті оновлення, ці системи стають вразливішими до збоїв. Наприклад, згідно з даними Lin та інших, система Microsoft Azure щодня зазнає збоїв приблизно у 0,1% серверних вузлів [13]. Такі збої можуть призвести до зниження доступності систем, фінансових втрат і погіршення досвіду користувачів. Дослідження IDC показують, що простій додатків може коштувати бізнесу до $550,000 за годину [17]. Ці значні втрати стимулюють потребу в автономних системах, які здатні самостійно керувати собою та усувати основні причини збоїв для підвищення якості та швидкості IT-послуг. Традиційні рішення для управління IT, що покладаються на експертні системи та механізми на основі правил, часто виявляються недостатньо адаптивними, ефективними та масштабованими. Такі підходи можуть ігнорувати актуальний стан системи в реальному часі, що призводить до неточних прогнозів та аналітики, заснованих на поточному стані системи. Крім того, вони опираються на традиційне інженерне мислення, яке акцентує увагу на ручному виконанні повторюваних завдань та аналізі окремих випадків, зокрема відтворенні помилок або аналізі логів [22]. Ці проблеми спонукали інтерес до заміни традиційних інструментів обслуговування на інтелектуальні платформи, здатні навчатися на великих обсягах даних та проактивно реагувати на інциденти. Організації все частіше звертаються до методів штучного інтелекту для менеджменту інформаційних технологій (Artificial Intelligence for IT Operations, AIOps) для попередження та усунення інцидентів, що мають значний вплив. Термін AIOps вперше був введений у 2017 році компанією Gartner, щоб охопити виклики, пов'язані з впровадженням штучного інтелекту (Artaficial Intelligence, AI) у процеси DevOps [20]. Спочатку цей термін походив від ITOA (аналітики IT-операцій), але з поширенням штучного інтелекту у різних галузях Gartner переосмислив його як AI для операційних систем. AIOps використовує технології великих даних та машинного навчання для інтелектуального вдосконалення, зміцнення та автоматизації різних IT-операцій. AIOps навчається на різноманітних даних, зібраних від сервісів, інфраструктур та процесів, і автономно вживає заходів для виявлення, діагностики та виправлення інцидентів у реальному часі.

Матеріали та методи досліджень. Ми зосередили увагу цього дослідження на досвіді компаній, що займаються розробкою AIOps рішень. Сьогодні організації розділяються на тих, хто займається розробкою системи для внутрішнього використання, і тих, хто впроваджує готові рішення від сторонніх розробників. Для цього ми скористалися наступними методами: створення проблеми, формування гіпотези, аналіз відкритих джерел, узагальнення і систематизація результатів наукової діяльності та практичної діяльності компаній. Додатково було проаналізовано досвід впровадження організацій першопроходьців в цьому напрямку [1, 3].

Як основну проблему дослідження ми визначили *відсутність стандарту, загальноприйнятого методу інтеграції існуючих рішень у процес*. Сьогодні можна спостерігати велику кількість наукових досліджень і програмних продуктів у сфері AIOps для вирішення різноманітних проблем. Сабхарвал та інші опублікували книгу «Практичний AIOps», у якій обговорюються практичні аспекти та впровадження AIOps [16]. Доступні кілька оглядів літератури з AIOps, які допомагають аудиторії краще зрозуміти цю сферу [15, 14]. Однак, більшість оглядів, пов'язаних зі штучним інтелектом, все ще залишаються тематичними, як-от виявлення аномалій за допомогою глибинного навчання, управління відмовами та аналіз першопричин. Наразі існує обмежена кількість досліджень, які надають цілісне уявлення про процес впровадження AIOps, охоплюючи ситуацію як у науковій спільноті, так і в промисловості. Проведені в даній статті дослідження заповнюють цю прогалину та зосереджують більше уваги на самому процесі розгортання інфраструктури AIOps. В результаті необхідно дати відповідь на наступні запитання:

- Що впливає на рішення організації щодо впровадження AIOps ?
- Які планувати очікування ?
- Якими є етапи процесу імплементації ?

Аналіз предметної області. Підходи, які застосовуються AIOps на сьогоднішній день, знаходяться на стадії активного розвитку. Нижче наведено короткий огляд кожного кроку менеджменту інцидентів і можливих рішень [2]:

1. Реєстрація інциденту – початковий етап життєвого циклу будь-якого інциденту. Великі дата-центри та хмарні послуги повинні мати проактивний моніторинг систем для вирішення інцидентів до того, як їх виявлять клієнти. Однак сьогодні більшість сповіщень налаштовуються відповідно до суворих правил, заснованих на раніше виявлених помилках і порогових значеннях. Використання AI для виявлення аномалій може ідентифікувати більшу кількість інцидентів до їх виникнення і до налаштування сповіщення. AI може ідентифікувати критичні компоненти системи та пріоритизувати помилки відповідно до їхнього впливу.

Існують різні підходи в ідентифікації і навіть передбаченні появи інциденту. Підходи до передбачення інцидентів є проактивними методами, спрямованими на запобігання збоям (або відмовам у крайніх

випадках) шляхом обробки як статичних аспектів, таких як вихідний код, так і динамічних аспектів, наприклад, доступності обчислювальних ресурсів. Основна мета полягає в тому, щоб запропонувати превентивні заходи або вжити негайних дій якомога раніше. Ці стратегії значно відрізняються залежно від запропонованої таксономії (тобто даних, що використовуються, сфери застосування тощо).

1.1. Прогнозування дефектів програмного забезпечення забезпечення (Software Defect Prediction, *SDP)* – це підхід, який використовується для оцінки ймовірності виникнення програмної помилки в функціональній одиниці коду, такій як функція, клас, файл або модуль. Основне припущення, що пов'язує SDP з виникненням збоїв, полягає в тому, що код із дефектами призводить до помилок і збоїв під час виконання. Традиційно програмне забезпечення, схильне до дефектів, ідентифікується за допомогою метрик коду, які використовуються для побудови предикторів дефектів. Нагаппан та інші запропонували підхід SDP на основі метрик складності коду [11] ще в 2010 році. В іншому дослідженні 2020 року Сюй та ін. [5] запропонували підхід для характеристики програмних дефектів, використовуючи дефектні піддерева в абстрактних синтаксичних деревах (Abstract Syntax Trees, AST). Цей підхід включає інформацію про зміни, що викликають виправлення коду. Спочатку розробляється тематична модель для узагальнення функціональних концепцій, пов'язаних із дефектами. Кожен вузол у дефектних піддеревах збагачений атрибутами, такими як типи, зміни, що викликають виправлення, та концепції коду. Після цього використовується класифікатор на основі графових нейронних мереж (Graph Neural Networks, GNN), де піддерева представлені як орієнтовані ациклічні графові структури. В 2022 Уддін та ін. [10] використали модель двоспрямованої довгої короткочасної пам'яті (Bidirectional Long Short-Term Memory Network, BiLSTM) разом із підходом до семантичних ознак на основі двоспрямованих кодувальних представлень з трансформерів (Bidirectional Encoder Representations from Transformers, BERT) для прогнозування дефектів у програмному забезпеченні. Ця комбінація захоплює семантичні ознаки коду, витягуючи контекстну інформацію з векторів токенів, які були навчені моделлю BERT за допомогою BiLSTM. Також інтегровано механізм уваги, який захоплює найбільш важливі ознаки для прогнозування. Ця методологія була покращена за рахунок техніки розширення даних, яка генерує додаткові тренувальні дані. Оцінка включає експерименти як із прогнозування дефектів у межах проєкту, так і з прогнозування дефектів між проєктами.

1.2. Прогнозування відмов апаратного забезпечення. У масштабних обчислювальних інфраструктурах проблема забезпечення надійності апаратного забезпечення є ключовою для досягнення цілей доступності послуг. Однак, через велику кількість компонентів і необхідність використання загальнодоступного апаратного забезпечення в дата-центрах, відмови апаратного забезпечення створюють значні виклики. Наприклад, компанія Google повідомляє, що 20-57% дисків мають принаймні одну помилку сектора протягом 4-6 років [8]. Жорсткі диски є найбільш часто замінюваними компонентами в хмарних обчислювальних системах і є однією з основних причин збоїв серверів. Щоб вирішити цю проблему, виробники жорстких дисків впровадили технології самостійного моніторингу, такі як метрики на базі технології самоконтролю, аналізу й звітування (Self Monitoring Analysis and Reporting Technology, SMART) у своїх пристроях зберігання. У підході, запропонованому Чжао та ін., використовуються приховані марковські та напівмарковські моделі для оцінки ймовірних послідовностей подій на основі спостережень метрик SMART із набору даних приблизно 300 дисків, дві третини яких були справними [21]. У дослідженні 2020 року Халіл та ін. [9] представили підхід для прогнозування потенційних апаратних відмов, спричинених старінням або зміною стану в ланцюгах передачі сигналів. Підхід використовує швидке перетворення Фур'є (Fast Fourier Transform, FFT) для виявлення частотних сигнатур відмов, застосовує аналіз головних компонент (Principal Component Analysis, PCA) для зменшення розмірності даних і виділення важливих характеристик, а також використовує згорткову нейронну мережу (Convolutional Neural Network, CNN) для навчання та класифікації відмов. Ця робота є особливо помітною, оскільки вперше вирішує проблему прогнозування відмов на рівні транзисторів для апаратних систем, охоплюючи відмови через старіння, коротке замикання та відкриті ланцюги.

1.3. Прогнозування відмов програмного забезпечення зосереджене на виявленні можливих збоїв на різних рівнях додатків, таких як процеси, задачі, віртуальні машини (Virtual Machine, VM), контейнери або вузли. Відомі підходи переважно ґрунтуються на аналізі системних метрик, станів сервісів, трасувань, логів і топологій. Одним із прикладів є підхід, запропонований Коеном та ін. [4], який використовує Баєсові мережі для виявлення зв'язків між спостережуваними змінними та станами сервісів. Ця стратегія дозволяє передбачати та запобігати порушенням цілей обслуговування (Service Level Objective, SLO) та збоям веб-сервісів. Система відстежує ключові метрики, такі як час процесора, читання з диска, використання простору підкачки, та інші, створюючи модель, яка описує складні залежності між цими метриками. Оптимальна структура графа, яка охоплює найбільш релевантні вхідні метрики, визначається за допомогою евристичного процесу відбору.

2. *Класифікація інциденту.* Після реєстрації важливо визначити постраждалу область і призначити квиток (ticket) компетентному інженеру або команді. Багато порушень договору про рівень обслуговування (Service Level Agreement, SLA) трапляються через те, що інцидент залишається непризначеним протягом тривалого часу. Існують рішення на основі AI, які можуть поєднувати машинне навчання та правила для призначення інцидентів на основі раніше вирішених запитів, поточного завантаження і доступності. Численні підходи на основі даних були запропоновані для оптимізації процесу призначення інцидентів, який також називається маршрутизацією або триажем (triage – розподіл), автоматично призначаючи інциденти відповідній сервісній команді чи особі. Зазвичай ці підходи передбачають навчання класифікатора на основі історичних звітів про інциденти, які містять текстову інформацію, дані про топологію або пріоритетні оцінки. Навчений класифікатор використовується для призначення нових інцидентів. Попередні роботи здебільшого покладалися на методи попередньої обробки текстів та моделі класифікації в рамках традиційного машинного навчання і статистичних підходів. Нещодавно увага зосередилася на сучасних методах обробки природної мови (Natural Language Processing, NLP) у поєднанні з глибоким навчанням. Наприклад, Лі та ін. [18] використали CNN і попередньо навчені вбудовування Word2Vec для призначення інцидентів. У роботі [7] процес триажу за допомогою глибинного навчання клітинного типу (Cell-type Deep Learning, DeepCT) представлено як безперервний процес, що включає інтенсивні обговорення між інженерами, використовуючи модель керованих рекурентних блоків (Gated Recurrent Units, GRU) з масковою стратегією уваги для поетапного оновлення результатів триажу на основі знань з обговорень.

3. *Пріоритизація інциденту*. Процес пріоритезації слід виконувати на основі впливу та угоди SLA з клієнтом. Сучасні технології NLP можуть виділяти факти та виконувати аналіз настроїв для встановлення правильного пріоритету. Людське втручання та перевірка зазвичай все ще потрібні, але завдяки автоматизації цей процес можна значно мінімізувати. Хен та ін. [6] провели масштабний емпіричний аналіз інцидентів в реальних онлайн-сервісних системах. Їхні результати підкреслюють категорію інцидентів під назвою «випадкові інциденти» (Incidental Incidents), які часто вважаються менш значущими та не пріоритизуються для негайного вирішення. Для розв'язання цієї проблеми автори запропонували пріоритезацію інцидентів за допомогою глибинного навчання (DeepIP, Deep learning-based Incident Prioritization) – систему, що використовує CNN на основі механізму уваги для ідентифікації та пріоритезації випадкових інцидентів, ґрунтуючись на їх історичних описах та інформації про топологію системи.

4*. Розслідування інциденту*. Залежно від категорії інциденту та інфраструктури програми, інженер вивчає можливі журнали, конфігурації, дані тощо. Кожне джерело розслідування слід розглядати окремо, залежно від підтримуваного додатка. Об'єднання процесу розслідування в один інтерфейс на основі AI є перспективною сферою для подальших досліджень та розробок. Помічники на основі AI можуть навчатися на попередніх інцидентах і робити висновки на основі отриманих даних, однак у більшості випадків для діагностики інциденту потрібне втручання людини, оскільки критерії можуть надходити з різних джерел. На сьогоднішній день дослідження сфокусовані на аналізі окремих рівнів інфраструктури, таких як база даних, мережеве з'єднання, програний код, хмарне середовище.

5. *Вирішення інциденту* – дії, що виконуються для відновлення нормального функціонування системи, усуваючи вплив інциденту. Подібно до попередніх етапів, участь людини необхідна через складність підтримуваних систем і різноманітні нетехнічні фактори, які потрібно враховувати, такі як можливість людських помилок, поточні відносини з клієнтом, інформація з джерел, над якими відсутній процес моніторингу. У дослідженні [19] запропоновано алгоритми на основі схожості для генерації рішень повторюваних проблем на основі інцидентних запитів. Цей підхід передбачає використання методу найближчих сусідів (k-Nearest Neighbors, k-NN) для отримання варіантів вирішення інциденту. Схожість між квитками оцінюється за допомогою комбінації числових, категоріальних та текстових даних з визначеними індивідуальними та агрегованими показниками схожості. Рішення було розширене для врахування хибнопозитивних квитків як в історичних даних, так і в поточних, за допомогою класифікації квитків за допомогою бінарного класифікатора і зважування їх важливості на основі прогнозів. Фінальна рекомендація щодо вирішення квитків враховує як важливість, так і схожість. Важливим результатом є покращення методів вилучення ознак, зокрема виявлення тем та навчання метрик для підвищення ефективності рекомендацій.

6. *Визначення та виконання запобіжних дій*. Для ефективного функціонування системи для кожного інциденту повинні бути визначені та виконані запобіжні дії. Для цього головна причина повинна бути чіткою і точною. Через складність і різноманітність джерел і факторів це наразі завдання, що вимагає когнітивних навичок людини, тому це перспективна область для подальших досліджень рішень на основі AI. У нещодавньому дослідженні був представлений підхід eWarn [12], який спрямований на онлайн-сервісні системи, що використовує історичні дані та інформацію про реальні тривожні сигнали для прогнозування ймовірності майбутніх інцидентів. Він поєднує такі інноваційні техніки:

– ефективна інженерія ознак для представлення відповідних шаблонів тривог;

– інтеграція багатоприкладного навчання (multi-instance learning), щоб мінімізувати вплив несуттєвих тривог;

– генерація зрозумілих звітів за допомогою техніки пояснень (Local Interpretable Model-Agnostic Explanations, LIME).

Цей підхід дозволяє не тільки передбачати інциденти, але й надавати пояснення для прийняття більш обґрунтованих рішень щодо інцидент-менеджменту. Таким чином, подібні методи дозволяють не лише передбачати можливі відмови систем, але й допомагати з аналізом причин і пропонувати превентивні заходи для їх уникнення.

Розробка методології імплементації методів AIOps.

В цьому дослідженні ми визначили, які інструменти існують вже сьогодні для реалізації автоматизацій етапів процесу вирішення інцидентів. Однак, питання їх впровадження в бізнес процес залишається відкритим. У науковій літературі описані різні підходи у використанні тих чи інших алгоритмів нейронних мереж, проте відсутня загальна схема впровадження технології відповідно до потреб організації.

Враховуючи ромаїття підходів, ми пропонуємо наступну схему впровадження AIOps (рис. 1):

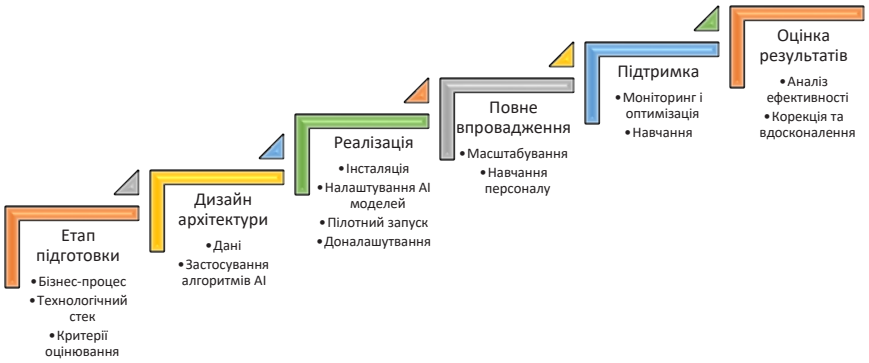

Рис. 1. Етапи впровадження AIOps

Проаналізуємо більш детально етапи імплементації AIOps у бізнес-процеси:

1. Етап підготовки

a) Аналіз бізнес процесу. Тут необхідно визначити стадії процесу, які будуть піддаватися AI автоматизації. Проводиться аналіз існуючих бізнес-процесів та IT-інфраструктури, щоб визначити, які саме операції найбільше потребують автоматизації або підвищення ефективності.

b) Технологічний стек. Це може бути внутрішня розробка, використання готових рішень або гібридне використання обох підходів.

c) Критерії оцінювання – метрики, які мають покращитися в результаті успішного впровадження. Встановлюються ключові показники ефективності (Key Performance Indicators, KPI), наприклад, час на відновлення після інциденту, середній час реакції на інциденти тощо.

2. Дизайн архітектури AIOps

a) Дані. Відповідно до стадії процесу, різні дані будуть піддаватися обробці алгоритмами AI. Проте визначення структури цих даних, способу нормалізації і місця зберігання є першим кроком у побудові AI-інфраструктури. Визначаються всі джерела даних, такі як журнали подій, моніторинг ресурсів, бази даних. Це можуть бути сервери, мережеві пристрої, хмарні сервіси.

b) Застосування алгоритмів AI. Розробляються моделі машинного навчання для моніторингу і прогнозування інцидентів. Це можуть бути моделі для виявлення аномалій або прогнозування інцидентів на основі минулих даних.

3. Реалізація

a) Інсталяція платформ для моніторингу та аналізу. Вибирається і встановлюється відповідне програмне забезпечення для управління та обробки великих обсягів даних. Інструменти на кшталт Prometheus, Splunk, чи ELK використовуються для моніторингу.

b) Налаштування AI-моделей. Інтеграція моделей AI та машинного навчання для обробки зібраних даних і виявлення аномалій в реальному часі.

c) Пілотне впровадження. Проводиться тестування системи на окремих частинах інфраструктури, щоб переконатися у правильності збору та аналізу даних.

d) Доналаштування алгоритмів. Налаштування моделей машинного навчання для більш точного прогнозування, виходячи з результатів тестування.

4. Впровадження на рівні всієї компанії

a) Масштабування. Після успішного тестування і налаштування на обмеженій кількості проектів система впроваджується у всі бізнес-процеси та IT-середовища компанії, що стосуються менеджменту інцидентів.

b) Навчання персоналу. Проводяться тренінги для співробітників з використання нових інструментів і процедур для взаємодії з системою AIOps.

5. Операційна підтримка

a) Моніторинг і оптимізація. Проводиться постійний моніторинг результатів роботи AIOps для вдосконалення моделей і підвищення ефективності.

b) Реалізація навчання на основі інцидентів. Використання зворотного зв'язку від інцидентів для навчання моделей AI, покращення прогнозування і швидкості реакції.

6. Оцінка результатів

a) Аналіз ефективності. Проводиться регулярний аналіз ефективності впровадженої системи на основі заздалегідь визначених метрик.

b) Корекція та вдосконалення. На основі аналізу результатів впровадження за ключовими метриками проводиться корекція процесів для забезпечення постійного вдосконалення.

Ця методологія може варіюватися в залежності від специфіки компанії або індустрії, але загальний підхід залишиться незмінним: від підготовки до масштабного впровадження і постійної оптимізації.

Висновки. Для підвищення ефективності процесу управління інцидентами ключовою є покрокова методологія впровадження в бізнес-процеси автоматизацій на базі штучного інтелекту з розгортанням інфраструктури AIOps. Досвід пілотних імплементацій таких технологій у галузі IT дав можливість систематизувати існуючі підходи відповідно до стадій процесу менеджменту інцидентів. Дана методологія може бути використана в організаціях і адаптована відповідно до конкретних потреб і цілей, адже враховує критерії оцінювання відповідно до вибраних KPI і постійне вдосконалення, що забезпечить максимальну адаптацію. Подальші дослідження є необхідними в цій галузі для визначення найкращих практик і створення стандартів, оскільки впровадження AI це процес із яким уже зіштовхуються більшість технологічних організацій. Застосовування розробленої методології і проведення кейс-стаді із визначенням можливих особливостей процесів організацій, є перспективним напрямком досліджень у галузі.

Список використаних джерел:

1. Databricks. Effective AIOps with Open Source Software in a Week. Youtube, 2021. URL: https://www.youtube.com/watch?v=NuL1u_CIkQw (дата звернення: 20.11.2024).

2. Diachenko Maksym, Roskladka Andrii. 2023. Approaches in managing IT services and implementation of AI automations. X INTERNATIONAL CONFERENCE Information Technology and Implementation (Satellite).C. 233.

3. Google Cloud Tech. Build an AIOps platform at enterprise scale with Google Cloud. Youtube. 2023 URL: https://www.youtube.com/watch?v=UdVaexipP6w (дата звернення: 20.11.2024).

4. Ira Cohen, Jeffrey S Chase, Moises Goldszmidt, Terence Kelly, and Julie Symons. 2004. Correlating Instrumentation Data to System States: A Building Block for Automated Diagnosis and Control. In OSDI, Vol. 4. 16–16. URL: https://dl.acm.org/doi/10.5555/1251254.1251270 (дата звернення: 20.11.2024).

5. Jiaxi Xu, Fei Wang, Jun Ai. Defect prediction with semantics and context features of codes based on graph representation learning. IEEE Transactions on Reliability 70. 2020. 613–625. URL: https://doi.org/10.1109/TR.2020.3040191 (дата звернення: 20.11.2024).

6. Junjie Chen, Shu Zhang, Xiaoting He, Qingwei Lin, Hongyu Zhang, Dan Hao, Yu Kang, Feng Gao, Zhangwei Xu, Yingnong Dang, et al. 2020. How incidental are the incidents? characterizing and prioritizing incidents for largescale online service systems. In Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering. C. 373–384. URL: https://doi.org/10.1145/3324884.3416624 (дата звернення: 20.11.2024).

7. Junjie Chen, Xiaoting He, Qingwei Lin, Hongyu Zhang, Dan Hao, Feng Gao, Zhangwei Xu, Yingnong Dang, Dongmei Zhang. 2019. Continuous incident triage for large-scale online service systems. 34th IEEE/ACM International Conference on Automated Software Engineering (ASE). IEEE. C. 364–375. URL: https://doi.org/10.1109/ASE.2019.00042 (дата звернення: 20.11.2024).

8. Justin Meza, Qiang Wu,Sanjev Kumar, and Onur Mutlu. 2015. A large-scale studyof flash memory failures in the field. ACM SIGMETRICS Performance Evaluation Review 43, 1. C.177–190. URL: https://doi.org/10.1145/2796314.2745848 (дата звернення: 20.11.2024).

9. Kasem Khalil, Omar Eldash, Ashok Kumar, Magdy Bayoumi. 2020. Machine learning-based approach for hardware faults prediction. IEEE Transactions on Circuits and Systems I: Regular Papers 67, 11. C. 3880–3892. URL: https://doi.org/10.1109/TCSI.2020.3010743 (дата звернення: 20.11.2024).

10. Md Nasir Uddin, Bixin Li, Zafar Ali, Pavlos Kefalas, Inayat Khan, Islam Zada. 2022. Software defect prediction employing BiLSTM and BERT-based semantic feature. Soft Computing 26, 16. C.7877–7891. URL: https://doi.org/10.1007/s00500-022-06830-5 (дата звернення: 20.11.2024).

11. Nagappan Nachiappan, Ball Thomas, Zeller Andreas. 2006. Mining metrics to predict component failures. In Proceedings of the 28th international conference on Software engineering. C. 452–461. URL: http://dx.doi.org/10.1145/1134349 (дата звернення: 20.11.2024).

12. Nengwen Zhao, Junjie Chen, Zhou Wang, Xiao Peng, Gang Wang, Yong Wu, Fang Zhou, Zhen Feng, Xiaohui Nie, Wenchi Zhang, et al. 2020. Real-time incident prediction for online service systems. In Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering.C. 315–326. URL: https://doi.org/10.1145/3368089.3409672 (дата звернення: 20.11.2024).

13. Qingwei Lin, Ken Hsieh, Yingnong Dang, Hongyu Zhang, Kaixin Sui, Yong Xu, Jian-Guang Lou, Chenggang Li, Youjiang Wu, Randolph Yao, et al. 2018. Predicting node failure in cloud service systems. In Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. C. 480–490. URL: https://doi.org/10.1145/3236024.3236060 (дата звернення: 20.11.2024).

14. Reiter Lena. 2021. AIOps – A Systematic Literature Review. Seminar IT-Management in the Digital Age. FH Wedel, Germany. URL: https://www.fh-wedel.de/fileadmin/Mitarbeiter/Records/Reiter_2021_-_AIOps_-_A_Systematic_Literature_Review.pdf (дата звернення: 20.11.2024).

15. Remil Youcef, et al. 2024. Aiops solutions for incident management: Technical guidelines and a comprehensive literature review. URL: https://arxiv.org/abs/2404.01363 (дата звернення: 20.11.2024).

16. Sabharwal N. 2022. Hands-on AIOps. Springer.

17. Stephen Elliot. 2014. Dev Ops and the cost of downtime: Fortune 1000 best practice metrics quantified. International Data Corporation (IDC).

18. Sun-Ro Lee, Min-Jae Heo, Chan-Gun Lee, Milhan Kim, Gaeul Jeong. 2017. Applying deep learning based automatic bug triager to industrial projects. In ESEC/FSE. ACM. C.926–931. URL: https://doi.org/10.1145/3106237.3117776 (дата звернення: 20.11.2024).

19. Wubai Zhou, Liang Tang, Chunqiu Zeng, Tao Li, Larisa Shwartz, and Genady Ya Grabarnik. 2016. Resolution recommendation for event tickets in service management. IEEE Transactions on Network and Service Management 13, 4 (2016), 954–967. URL: https://doi.org/10.1109/INM.2015.7140303 (дата звернення: 20.11.2024).

20. Xianping Quand Jingjing Ha. 2017. Next generation of devops: Aiops in practice. SREcon17.

21. Ying Zhao, Xiang Liu, Siqing Gan, and Weimin Zheng. 2010. Predicting disk failures with HMM-and HSMM-based approaches. In Advances in Data Mining. Applications and Theoretical Aspects: 10th Industrial Conference, ICDM 2010, Berlin, Germany, July 12-14. Proceedings 10. Springer. C. 390–404. URL: https://doi.org/10.1007/978-3-642-14400-4_30 (дата звернення: 20.11.2024).

22. Yingnong Dang, Qingwei Lin, and Peng Huang. AIOps: real-world challenges and research innovations. In 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion). IEEE, 2019. C. 4–5. URL: https://doi.org/10.1109/ICSE-Companion.2019.00023 (дата звернення: 20.11.2024).