ISSN 2079-5459 (print)
ISSN 2413-4295 (online)

УДК 004.738:620.179 doi:10.20998/2413-4295.2025.01.05

СИСТЕМА ЗБОРУ СТАТИСТИЧНИХ ДАНИХ ПРО ДЕФЕКТИ МЕТАЛОКОНСТРУКЦІЙ НА ОСНОВІ TELEGRAM BOT

Л. В. ЗАЙЦЕВА*, Д. О. МАКАРОВ

кафедра комп'ютерних та радіоелектронних систем контролю та діагностики, НТУ «ХПІ», Харків, УКРАЇНА
**e-mail:Liliia.zaitseva@khpi.edu.ua*

АНОТАЦІЯ *Проаналізовано роль чат-ботів у процесах автоматизації, зокрема для збору даних про дефекти у виробничих системах. Розглянуто технологічні аспекти створення чат-бота для моніторингу якості продукції, його переваги та функціональні можливості. Показано, що чат-боти спрощують введення даних, знижують витрати часу на рутинні операції та мінімізують помилки користувачів. Функціональність бота дає змогу видаляти записи, що дозволяє користувачам управляти своїми даними, а також отримувати звіти про кількість дефектів, їх типи та динаміку змін у часі. Інформація про дефекти зберігається у зручному для аналізу вигляді. Це дозволяє здійснювати швидкий доступ до даних і проводити детальний аналіз інформації про дефекти, що в свою чергу дає змогу приймати обґрунтовані рішення щодо покращення процесів виробництва.. Розглянуто застосування платформ Telegram, Node.js та бази даних MongoDB у процесі розробки. Акцентовано увагу на простоті інтеграції чат-ботів із системами управління підприємством. У даного бота є можливість інтеграції з іншими системами. Це означає, що дані, зібрані через бот, можуть бути автоматично передані до системи управління якістю, що зменшує ймовірність помилок, пов'язаних із ручним введенням даних. Завдяки цьому підприємство може швидше реагувати на виявлені дефекти та вживати заходів для їх усунення. Підтверджено ефективність використання чат-бота для автоматизації аналізу дефектів, зменшення навантаження на співробітників та формування статистичних звітів. Обґрунтовано доцільність використання чат-ботів для підвищення продуктивності виробництва та задоволеності клієнтів. Спеціальні статистичні звіти, які формуються ботом, дають змогу керівництву отримувати інформацію про типи дефектів, частоту їх виникнення та інші важливі параметри Підтверджено, що інтерактивний інтерфейс дозволяє користувачам швидко взаємодіяти із системою, а автоматизоване оброблення даних сприяє ухваленню ефективних управлінських рішень. Результати роботи демонструють потенціал чат-ботів для вдосконалення процесів управління якістю та оптимізації ресурсів у сучасному конкурентному середовищі.*
Ключові слова*: чат-бот; дефект; якість; оптимізація; база даних.*

A SYSTEM FOR COLLECTING STATISTICAL DATA ABOUT DEFECTS IN METAL STRUCTURES BASED ON TELEGRAM BOT

L. ZAITSEVA*, D. MAKAROV

Department of Computer and Radio-electronic Systems for Testing and Diagnostics, NTU "KhPI", Kharkiv, UKRAINE

ABSTRACT *The role of chatbots in automation processes, in particular, for collecting data on defects in production systems is analyzed. The technological aspects of creating a chatbot for product quality monitoring, its advantages and functionalities are considered. It is shown that chatbots simplify data entry, reduce time spent on routine operations, and minimize user errors. The bot's functionality allows deleting records, which allows users to manage their data and receive reports on the number of defects, their types, and the dynamics of changes over time. Information about defects is stored in an easy-to-analyze form. This allows for quick access to data and detailed analysis of defect information, which in turn allows for informed decisions to improve production processes. The use of Telegram, Node.js, and MongoDB platforms in the development process is considered. The author emphasizes the ease of integration of chatbots with enterprise management systems. This bot has the ability to integrate with other systems. This means that the data collected through the bot can be automatically transferred to the quality management system, which reduces the likelihood of errors associated with manual data entry. As a result, the company can respond to detected defects faster and take measures to eliminate them. The effectiveness of using a chatbot to automate defect analysis, reduce the workload of employees, and generate statistical reports is confirmed. The expediency of using chatbots to increase production productivity and customer satisfaction is substantiated. Special statistical reports generated by the bot allow the management to receive information about the types of defects, their frequency, and other important parameters It is confirmed that the interactive interface allows users to quickly interact with the system, and automated data processing contributes to effective management decisions. The results demonstrate the potential of chatbots to improve quality management processes and optimize resources in today's competitive environment.*
Keywords*: chatbot; defect; quality; optimization; database.*

Вступ

Чат-боти, які представляють собою комп'ютерні програми для автоматизації комунікації, набирають популярності в технічних сферах завдяки своїй здатності швидко і ефективно виконувати специфічні завдання. Вони дозволяють зменшити витрати часу на рутинні процеси, такі як введення даних, контроль якості та технічна підтримка. Актуальність використання чат-ботів у технічних задачах зумовлена зростаючою потребою в автоматизації, яка дозволяє оптимізувати роботу

© Л. В. ЗАЙЦЕВА, Д. О. МАКАРОВ, 2025

підприємств, підвищувати продуктивність та зменшувати ймовірність людських помилок [1-3].

Однією з основних переваг чат-ботів є їх зручність у використанні. На відміну від традиційних програм і додатків, які потребують установки та навчання, чат-боти доступні через популярні платформи, такі як Telegram, що робить їх доступними для більш широкої аудиторії. Інтерфейс чат-ботів інтуїтивно зрозумілий — користувачі взаємодіють з ними, використовуючи прості команди або кнопки, що зменшує час на навчання та використання.

Крім того, чат-боти можуть обробляти запити одночасно від багатьох користувачів, забезпечуючи оперативність реагування. Це особливо важливо в умовах підвищеної завантаженості або обмежених ресурсів. Чат-боти також можуть легко інтегруватися з системами управління, що дозволяє автоматизувати рутинні завдання, такі як введення даних і звітність, зменшуючи ризик помилок.

Мета роботи

Основною метою даної роботи є розробка чат-бота для збору статистичної інформації про дефекти, що спростить і автоматизує процеси моніторингу якості продукції. Додатково, дана розробка передбачає створення зручного інтерфейсу для користувачів, який дозволяє швидко та ефективно фіксувати дефекти. Важливою складовою є аналіз отриманих даних, що допоможе у прийнятті рішень для покращення якості продукції [4-7]. Таким чином, реалізація цього проекту сприятиме підвищенню ефективності управління якістю на підприємствах.

Виклад основного матеріалу

У сучасному світі, де швидкість обробки інформації та ефективність комунікацій стають критично важливими для успіху виробництва, чат-боти набувають все більшої популярності. Чат-бот — це комп'ютерна програма, що функціонує на основі заздалегідь заданих алгоритмів, машинного навчання або нейромереж, дозволяючи здійснювати комунікацію з користувачами в режимі реального часу через текстовий або голосовий інтерфейс. Вони широко використовуються в різноманітних сферах, включаючи обслуговування клієнтів, маркетинг, медицину, фінансові послуги та багато інших. Основна мета чат-ботів полягає у виконанні рутинних завдань, таких як надання довідкової інформації, виконання розрахунків, управління замовленнями, а також автоматизація процесів комунікації з клієнтами.

Зростання популярності систем обміну миттєвими повідомленнями, таких як Telegram, WhatsApp та інших, спростило інтеграцію чат-ботів у повсякденне життя. Чат-боти в Telegram, наприклад, мають зручне меню та вбудовані кнопки, що дозволяє користувачам без зайвих зусиль взаємодіяти з ботом та отримувати необхідну інформацію. З технологічного боку, чат-боти можуть обробляти велику кількість запитів одночасно, що робить їх незамінними у масштабних проектах. В Україні, з розвитком інформаційних технологій, компанії починають усвідомлювати важливість цих інструментів, що веде до збільшення попиту на розробку чат-ботів.

Актуальність чат-ботів в Україні особливо зростає, оскільки промисловість поступово переходить до автоматизації своїх процесів. Це не лише полегшує роботу співробітникам, а й підвищує задоволеність клієнтів, оскільки користувачі отримують швидкі відповіді на свої запитання без потреби чекати на допомогу оператора. Наприклад, чат-боти можуть значно зменшити навантаження на службу підтримки, автоматизуючи первинні запити клієнтів. Відповідно до статистичних даних, чат-боти можуть скоротити час і витрати на оформлення звітів з неруйнівного контролю, що є суттєвим аргументом для їх впровадження.

При розробці чат-бота для збору та обробки статистичної інформації про дефекти використовуються сучасні технології, такі як Node.js, Visua Studio Code та MongoDB. Node.js є потужною платформою з відкритим кодом, що дозволяє створювати високопродуктивні мережеві додатки, засновані на JavaScript. Ця платформа дозволяє створювати чат-боти, які можуть ефективно обробляти численні запити одночасно, що є не менш важливим. Visual Studio Code, популярний текстовий редактор, забезпечує зручний інтерфейс для розробки, пропонуючи безліч корисних функцій, таких як інтеграція з системами контролю версій, налагодження коду та підсвічування синтаксису. MongDB, у свою чергу, є потужною системою управління базами даних, яка підтримує документо-орієнтовану модель даних. Це дозволяє зберігати та обробляти великі обсяги інформації без необхідності визначати структуру таблиць, що робить її ідеальним вибором для проектів, які вимагають гнучкості у зберіганні даних [8].

Node.js - програмна платформа на платформі V8, яка транслює JavaScript-код в машинний. Цей двигун перетворює вузькоспеціалізовану мову JavaScript на мову загального призначення. У Node.js з'явилася можливість взаємодіяти з пристроями вводу-виводу через свій API, написаний на C++, використовувати сторонні бібліотеки, створені за допомогою різних мов програмування, викликаючи їх з JavaScript-коду.

Node.js здебільшого використовує як back-end мову програмування, виконуючи функцію сервера. Водночас існує можливість створення десктопних віконних додатків за допомогою таких фреймворків, як NW.js, AppJS чи Electron. Однією з унікальних характеристик Node.js є її придатність для розробки програм у сфері Інтернету речей (Internet of Things,

ISSN 2079-5459 (print)
ISSN 2413-4295 (online)

IoT). Смарт-браслети, колонки та інші «розумні» пристрої, які стають дедалі популярнішими, потенційно можуть бути запрограмовані з використанням Node.js.

Розвиток Node.js тісно пов'язаний із створенням нового JavaScript-движка V8, що був розроблений у данському підрозділі компанії Google. Головний інженер Lars Bak визначив ключові завдання: підвищення продуктивності та масштабованості. Перша версія V8 була завершена 3 липня 2008 року. Невдовзі світ побачив перший браузер Chromium, у якому був інтегрований цей новий движок. Крім браузерів на основі Chromium, таких як Chrome, Comodo Dragon, Opera, V8 застосовується й у ряді інших продуктів:

Maxthon — веб-браузер з вбудованим блокуванням реклами, що працює на рендеринг-движках WebKit і Trident;

стандартний браузер операційної системи Android;

операційна система Android;

HP webOS — ОС від Hewlett-Packard для смартфонів, нетбуків та планшетів;

Google Chrome OS — операційна система Google, розроблена на базі Chromium і орієнтована на хмарні сервіси.

У 2009 році програміст Rayn Dahl представив Node.js, що стало результатом двох років експериментів у сфері серверних веб-компонентів. Під час своїх досліджень він дійшов висновку, що подієво-орієнтована архітектура є більш ефективною за традиційну модель потокового паралелізму через її простоту, низькі витрати та високу швидкодію. Node.js запропонував новий підхід до створення масштабованих мережевих серверів. Його розробка здійснюється за підтримки компанії Joyent.

Як приклад на рис. 1 зображено фрагмент коду, який реалізує простий сервер.

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Рис. 1 - Реалізація простого сервера на Node.js

Bot API забезпечує HTTP-інтерфейс для роботи з ботами на платформі Telegram. Бот — це окремий обліковий запис, створений для автоматизованої обробки та надсилання повідомлень. Для отримання оновлень від бота існують два підходи, що відрізняються своєю логікою:

Long polling — програма регулярно надсилає запити до серверів Telegram для перевірки наявності нових оновлень. За замовчуванням період очікування становить 100 мс.

Webhook — сервери Telegram автоматично надсилають сповіщення програмі одразу після появи оновлень.

Оновлення, що надходять, зберігаються на сервері Telegram не довше 24 годин, якщо їх не оброблено. Незалежно від обраного методу, відповідь містить об'єкт Update, відформатований у JSON.

Запити до Telegram Bot API мають виконуватися через HTTPS у форматі: https://api.telegram.org/bot<token>/НАЗВА_МЕТОДУ. Принцип роботи взаємодії чат-бота та користувача зображено на рис. 2.

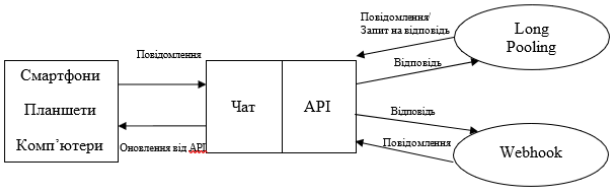

Рис. 2 - Принцип роботи чат-бота на платформі Telegram [9]

Для отримання токена необхідно звернутися до бота @BotFather. Основні методи API:

getUpdates — отримання оновлень за допомогою long polling;

setWebhook — прив'язка URL для оновлень до бота;

sendMessage — надсилання текстових повідомлень у Telegram;

sendLocation — передача геолокації в Telegram;

getFile — завантаження файлу за іменем.

API підтримує запити типу POST та GET. Передавати параметри можна через:

URL-запити;

формат application/x-www-form-urlencoded;

формат application/json (не використовується для передачі файлів);

multipart/form-data (підходить для передачі файлів).

Amazon Web Services (AWS) — одна з найпопулярніших платформ хмарних технологій у світі. Заснована у 2006 році, AWS стала першим комерційно доступним сервісом розподілених обчислень. Платформа дозволяє орендувати серверні ресурси Amazon і користуватися обчислювальною потужністю без необхідності купувати власне обладнання. Її надійність і широкий спектр послуг роблять AWS привабливим рішенням для великих корпорацій, медіакомпаній, державних установ і стартапів.

Серед клієнтів AWS — Bloomberg, Netflix, NFL, NASA та інші. Платформа пропонує такі переваги:

зручна програмна архітектура;

відсутність довгострокових зобов'язань;

централізоване управління та підтримка гібридних обчислень;

швидка установка й видалення програм;

економія витрат на власні сервери;

конкурентні ціни володіння.

MongoDB — документоорієнтована база даних з відкритим вихідним кодом, яка не вимагає попереднього опису схеми таблиць. Основні можливості MongoDB:

підтримка масштабованості та відмовостійкості (асинхронна реплікація, реплікаційні набори, розподіл бази даних на вузли);

зберігання даних у документальному форматі (JSON-подібні структури);

ефективна робота з великими об'єктами;

підтримка серверних функцій, адміністративний інтерфейс, використання Map/Reduce;

мова запитів — JavaScript;

профілювання запитів;

різноманітні атомарні операції (складний пошук, умовна вставка/оновлення тощо);

робота з різними типами даних, включно з масивами;

підтримка B-Tree індексів;

повнотекстовий пошук із морфологічною підтримкою;

журналювання змін у базі даних.

На відміну від традиційних реляційних баз даних, MongoDB використовує документоорієнтовану модель даних. Завдяки цьому база працює швидше, має більшу гнучкість у масштабуванні та є більш простою в роботі. Водночас вибір між MongoDB і реляційними базами залежить від конкретного завдання. У деяких випадках MongoDB забезпечує значну перевагу, наприклад, коли потрібно працювати зі складно структурованими даними. В інших ситуаціях доцільніше використовувати класичні реляційні бази даних. Крім того, можна комбінувати обидва підходи, зберігаючи один тип даних у MongoDB, а інший — у традиційних базах.

Основні характеристики MongoDB:

1. Формат даних

MongoDB зберігає дані у форматі BSON (Binary JSON), який за своєю структурою схожий на JSON. BSON забезпечує швидкий пошук і обробку даних, хоча займає більше місця порівняно з JSON. Однак підвищена швидкість роботи компенсує цей недолік.

2. Кросплатформність

MongoDB написана мовою C++, що дозволяє запускати її на різних операційних системах: Windows, Linux, MacOS, Solaris. Для зручності рекомендується використовувати офіційні бібліотеки, хоча також доступний вихідний код, який можна скомпілювати самостійно.

3. Колекції

У MongoDB дані організовані у колекції, які є аналогом таблиць у SQL. Однак, на відміну від таблиць, колекції можуть містити об'єкти з різною структурою та різними наборами властивостей.

4. Реплікація

Система реплікації MongoDB включає головний вузол і вторинні вузли. Вторинні вузли автоматично синхронізуються з головним, зберігаючи дані в актуальному стані. У разі збою головного вузла один із вторинних автоматично стає новим головним.

5. Простота використання

Відсутність жорсткої схеми дозволяє MongoDB легко адаптуватися до змін у структурі даних. Це спрощує роботу розробників, які можуть швидко масштабувати базу без необхідності її повної перебудови.

6. GridFS

Для зберігання великих об'єктів MongoDB використовує технологію GridFS, яка дозволяє зберігати файли розміром понад 16 МБ. Дані розділяються на сегменти (по 256 КБ) і зберігаються у двох колекціях: files для метаданих файлів і chunks для їхніх сегментів.

Visual Studio Code — це легкий і потужний текстовий редактор для розробки, сумісний із Windows, MacOS і Linux. Він пропонує вбудований відладчик, інструменти для роботи з Git, підсвічування синтаксису, рефакторинг коду та функцію IntelliSense для автозаповнення.

Редактор був представлений 29 квітня 2015 року компанією Microsoft на конференції Blink, а 18 листопада 2015 року випущений у реліз під ліцензією Массачусетського технологічного інституту (MIT).

Visual Studio Code побудований на основі фреймворку Electron, який дозволяє створювати десктопні програми з використанням Node.js і двигуна Blink. Хоча редактор базується на Electron, для реалізації його функціоналу використовується власний веб-редактор Monaco, розроблений для Visual Studio Online, а не редактор Atom.

Інтерфейс IDE Visual Studio Code представлений на рис. 3.

Діаграма розгортання розробленого чат-боту візуалізує ключові компоненти системи, зокрема сервер, на якому розташований чат-бот, а також зв'язки між ботом і базою даних MongDB, рис.4.

Ця діаграма демонструє, як користувач взаємодіє з ботом через інтерфейс Telegram, що забезпечує зручність та доступність сервісу. Опис бази даних зображено на рис. 5. База даних "defectlog" містить таблицю "records", в якій зберігається вся необхідна інформація про дефекти. Поля таблиці включають унікальний ідентифікатор запису, ідентифікатор користувача, назву виробу з дефектом, назву дефекту, статус дефекту (забракований чи ні) та примітки, які можуть містити важливу інформацію для подальшого аналізу.

ISSN 2079-5459 (print)
ISSN 2413-4295 (online)

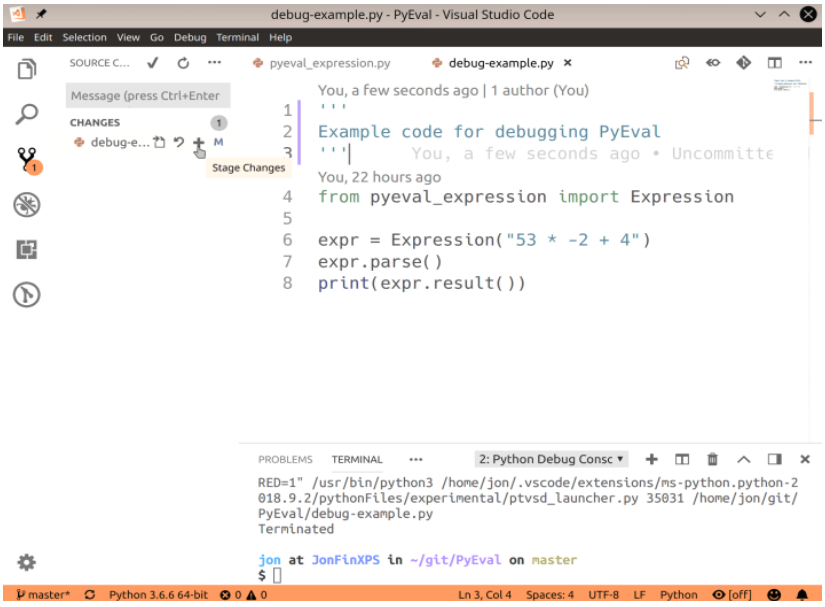

Рис. 3 - Інтерфейс IDE Visual Studio Code

Рис. 4 – Діаграма розгортання

Рис. 5 – Опис бази даних

Системні вимоги для запуску даного додатку є досить універсальними, що робить його доступним для широкого кола користувачів. Користувачі можуть працювати з додатком на різних операційних системах, таких як Windows, Linux, MacOS, ChromeOS, а також використовувати мобільні платформи, такі як Android та iOS. Головна умова – наявність доступу до Інтернету, що дозволяє чат-боту виконувати свої функції в реальному часі. Керівництво користувача надає детальні інструкції для початку роботи з ботом, включаючи процес активації, створення записів, перегляду статистики та отримання допомоги [10].

Користувачеві потрібно лише знайти бота за ім'ям або перейти за посиланням. Після активації, бот надає можливість створювати нові записи про дефекти, переглядати існуючі записи та статистику, а також взаємодіяти через зручний інтерфейс. Створення нового запису передбачає вибір назви виробу та типу дефекту, а також можливість введення додаткових приміток. Користувач може переглядати записи за різними критеріями, що дозволяє швидко знаходити потрібну інформацію. Початок роботи з ботом, меню «журналу записів» та як переглянути записи за критерієм зображено на рис. 6-8 відповідно. Важливою

функцією є можливість отримання статистики за певними періодами, що дозволяє аналізувати дані про дефекти та ухвалювати обґрунтовані рішення щодо покращення якості продукції [11].

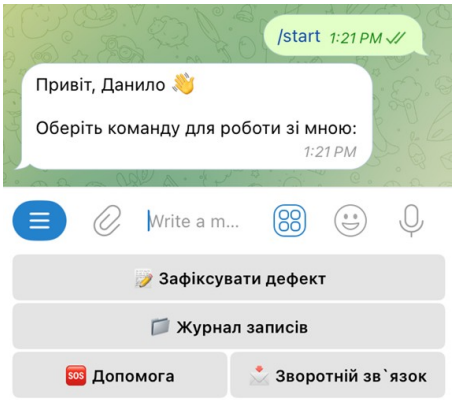

Рис. 6 – Меню чат-бота

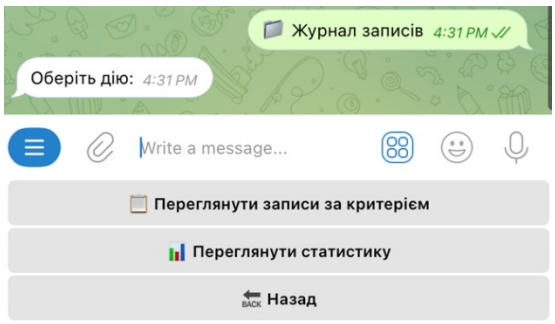

Рис. 7 – Меню вкладки «Журнал записів»

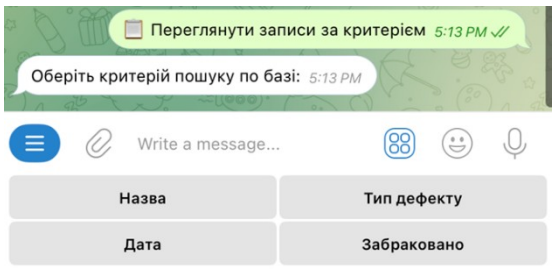

Рис. 8 – Меню вкладки «Переглянути записи за критерієм»

Функціональність бота включає в себе можливість видалення записів, що дозволяє користувачеві управляти своїми даними, а також отримання звітів про кількість дефектів, їх типи та динаміку змін у часі. Це надає змогу компаніям своєчасно реагувати на проблеми з якістю продукції, проводити аналіз причин дефектів і вживати заходів для їх усунення. Впровадження таких інструментів у виробництві дозволяє зекономити час і ресурси, підвищити ефективність управління якістю, а також створити більш прозору систему моніторингу якості продукції [12,13].

Завдяки своїй простоті у використанні та потужній функціональності, розроблений чат-бот не лише автоматизує рутинні завдання, але й сприяє підвищенню ефективності управління якістю на підприємствах. Впровадження таких інструментів дозволяє зекономити час і ресурси, покращити процеси збору та аналізу даних, а також підвищити рівень задоволеності користувачів. Результатом роботи чат-бота є зниження витрат на обслуговування клієнтів, підвищення їх лояльності, а також формування позитивного іміджу компанії в очах споживачів.

Обговорення результатів

З початку впровадження чат-бота виявлено кілька ключових переваг, які позитивно впливають на роботу підприємства. По-перше, автоматизація процесу збору даних значно зменшила час, необхідний для обробки інформації про дефекти. Це стало можливим завдяки інтерактивному інтерфейсу, який дозволяє користувачам швидко вводити необхідні дані через прості команди. Наприклад, функція створення нового запису займає всього кілька хвилин, у порівнянні з попередніми методами, що вимагали більше часу на заповнення документів.

По-друге, завдяки структурованій базі даних, яка використовується в MongDB, інформація про дефекти тепер зберігається у зручному для аналізу вигляді. Це дозволяє здійснювати швидкий доступ до даних і проводити детальний аналіз тенденцій по дефектах, що в свою чергу дає змогу приймати обґрунтовані рішення щодо покращення процесів виробництва. Спеціальні статистичні звіти, які формуються ботом, дають змогу керівництву отримувати інформацію про типи дефектів, частоту їх виникнення та інші важливі параметри, що є критично важливими для підвищення якості продукції.

Додатковою перевагою чат-бота є можливість інтеграції з іншими системами. Це означає, що дані, зібрані через бот, можуть бути автоматично передані до системи управління якістю, що зменшує ймовірність помилок, пов'язаних із ручним введенням даних. Завдяки цьому підприємство може швидше реагувати на виявлені дефекти та вживати заходів для їх усунення. Зокрема, швидке реагування на дефекти може значно знизити витрати на виправлення помилок і підвищити загальну продуктивність.

Ще однією важливою перевагою є зручність використання чат-бота. Користувачі отримують простий і зрозумілий інтерфейс, що дозволяє їм легко орієнтуватися в меню бота та швидко виконувати необхідні дії. Це знижує потребу в додаткових навчаннях або інструкціях, що, у свою чергу, економить час і ресурси підприємства.

Водночас, незважаючи на всі переваги, слід зазначити, що реалізація чат-бота потребує певних зусиль для підтримки та оновлення. Наприклад, необхідно регулярно перевіряти коректність роботи системи, оновлювати базу даних і адаптувати бот до змін у процесах виробництва. Однак, незважаючи на ці виклики, переваги, які надає чат-бот, очевидні і

ISSN 2079-5459 (print)
ISSN 2413-4295 (online)

значні. Технологічний прогрес у сфері автоматизації процесів робить подібні рішення все більш затребуваними та актуальними для підприємств, які прагнуть підвищити свою ефективність і конкурентоспроможність.

Отже, результати впровадження чат-бота для збору статистичної інформації про дефекти підтверджують, що інноваційні рішення можуть суттєво змінити підходи до управління якістю на підприємствах. Використання таких технологій не лише оптимізує внутрішні процеси, але й підвищує загальний рівень задоволеності споживачів, адже якість продукції безпосередньо впливає на репутацію компанії. Чат-бот, завдяки своїм можливостям, сприяє формуванню більш ефективних процесів контролю якості, що є ключовим фактором у сучасному конкурентному середовищі.

Висновки

У контексті реалізації чат-бота для збору статистичної інформації про дефекти, отримані результати демонструють значний потенціал технології автоматизації у сфері контролю якості.

По-перше, створення чат-бота забезпечило суттєве підвищення ефективності процесу збору даних. Зменшення часу на обробку запитів і збирання інформації безпосередньо вплинуло на можливості підприємства швидше реагувати на виявлені дефекти. Це стало можливим завдяки інтуїтивно зрозумілому інтерфейсу та автоматизації рутинних завдань, що в свою чергу дозволяє працівникам зосередитися на більш складних завданнях, пов'язаних із покращенням процесів.

По-друге, інтеграція чат-бота з системами управління якістю та базами даних підвищила точність і надійність інформації, що збирається. Документування даних про дефекти в реальному часі створює можливість для глибшого аналізу та виявлення тенденцій, що з часом дозволяє оптимізувати виробничі процеси. Таким чином, підприємство отримує не лише дані, а й стратегічну інформацію, необхідну для прийняття обґрунтованих управлінських рішень.

Третім важливим аспектом є вплив чат-бота на взаємодію між працівниками та системами управління якістю. Завдяки простоті користування, співробітники можуть швидше отримувати необхідну інформацію, що покращує комунікацію та взаємодію всередині команди. Це, в свою чергу, сприяє підвищенню морального духу та залученості працівників у процес покращення якості продукції.

Необхідно також зазначити, що реалізація чат-бота відкриває нові можливості для подальшого розвитку технологій в сфері контролю якості. З огляду на швидкий розвиток штучного інтелекту та машинного навчання, можна очікувати, що в майбутньому чат-боти зможуть не лише збирати, але й аналізувати дані, пропонувати рішення на основі отриманих результатів та навіть прогнозувати можливі дефекти. Це може значно зменшити витрати на контролювання якості та підвищити загальну продуктивність підприємства.

Отже, впровадження чат-бота для збору статистичної інформації про дефекти є важливим кроком у напрямку автоматизації процесів контролю якості на підприємствах. З отриманими результатами можна стверджувати, що такі рішення не лише спрощують повсякденні задачі, але й сприяють стратегічному розвитку організацій, підвищуючи їхню конкурентоспроможність на ринку. Успішний досвід впровадження чат-бота в даному проекті може стати прикладом для інших підприємств, які прагнуть модернізувати свої процеси та впроваджувати інновації у сфері управління якістю.

Список літератури

1. Modrzyk N. Building telegram bots: develop bots in 12 programming languages using the telegram bot API. Apress, 2018. doi: 10.1007/978-1-4842-4197-4.
2. Ismawati D., Prasetyo I. The development of Telegram bot through short story. *Brawijaya International Conference on Multidisciplinary Sciences and Technology* (BICMST 2020). Atlantis Press, 2020. P. 209-212. doi: 10.2991/assehr.k.201021.049.
3. Abd Halim I. H., Mahamad A. I., Fuzi M. F. M. Automated alert system for river water level and water quality assessment using telegram bot API. *Journal of Computing Research and Innovation.* 2021. Vol. 6. 3. P. 65-74. doi: 10.24191/jcrinn.v6i3.234.
4. Gupta M. et al. Advances in applications of Non-Destructive Testing (NDT): A review. *Advances in Materials and Processing Technologies*. 2022. Vol. 8. 2. P. 2286-2307. doi: 10.1080/2374068X.2021.1909332.
5. Wang B. et al. Non-destructive testing and evaluation of composite materials/structures: A state-of-the-art review. *Advances in mechanical engineering*. 2020. Vol. 12. 4. P. 1687814020913761. doi: 10.1177/1687814020913761.
6. Zaitseva L., Gorkunov B. Theoretical Aspects of Excitation and Reception of Acoustic Waves for Non-Destructive Testing of Electromechanical and Conversion Systems. *2023 IEEE 4th KhPI Week on Advanced Technology (KhPIWeek).* IEEE, 2023. P. 1-5. doi: 10.1109/KhPIWeek61412.2023.10312998.
7. Zaitsev R., Kirichenko M., Zaitseva L., Radoguz S. Automation measurement system of semiconductor devices parameters. *2020 IEEE 4th International Conference on Intelligent Energy and Power Systems*, IEPS 2020. Proceedings, 2020. P. 126–129. P. 9263136. doi: 10.1109/IEPS51250.2020.9263136.
8. Medvedyk A., Lohoida M., Rybchak Z., Kulyna O. IT Slang: Development of Telegram Chatbot. *In COLINS*. 2023. Vol. 2. 3. P. 152-162.
9. Левицький О. Л. Розробка телеграм боту для управління контентом рекламної агенції» : диплом бакалавра. ДУТ, Київ, 2021. 44 с. URL: https://duikt.edu.ua/repozitorii/ipz/2021/ППЗ-52/Левицький Олег.pdf (дата звернення: 23.02.2025).
10. Deng X., Yu Z. A meta-analysis and systematic review of the effect of chatbot technology use in sustainable education. Sustainability. 2023. Vol. 15. 4. P. 2940. doi: 10.3390/su15042940.

11. Luo B. et al. A critical review of state-of-the-art chatbot designs and applications. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. 2022. Vol. 12. 1. P. e1434. doi: 10.1002/widm.1434.
12. Silva M. I. et al. Review of conventional and advanced non-destructive testing techniques for detection and characterization of small-scale defects. *Progress in Materials Science*. 2023. Vol. 138. P. 101155. doi: 10.1016/j.pmatsci.2023.101155.
13. Shi P., Su S., Chen Z. Overview of researches on the nondestructive testing method of metal magnetic memory: Status and challenges. *Journal of Nondestructive Evaluation*. 2020. Vol. 39. Art. 43. doi: 10.1007/s10921-020-00688-z.

References (transliterated)

1. Modrzyk N. Building telegram bots: develop bots in 12 programming languages using the telegram bot API. *Apress*, 2018, doi:10.1007/978-1-4842-4197-4.
2. Ismawati D., Prasetyo I. The development of Telegram bot through short story. *Brawijaya International Conference on Multidisciplinary Sciences and Technology (BICMST 2020)*, Atlantis Press, 2020, pp. 209–212, doi:10.2991/assehr.k.201021.049.
3. Abd Halim I. H., Mahamad A. I., Fuzi M. F. M. Automated alert system for river water level and water quality assessment using telegram bot API. *Journal of Computing Research and Innovation*, 2021, vol. 6, no. 3, pp. 65–74, doi:10.24191/jcrinn.v6i3.234.
4. Gupta M., et al. Advances in applications of Non-Destructive Testing (NDT): A review. *Advances in Materials and Processing Technologies*, 2022, vol. 8, no. 2, pp. 2286–2307, doi:10.1080/2374068X.2021.1909332.
5. Wang B., et al. Non-destructive testing and evaluation of composite materials/structures: A state-of-the-art review. *Advances in Mechanical Engineering*, 2020, vol. 12, no. 4, pp. 1687814020913761, doi:10.1177/1687814020913761.
6. Zaitseva L., Gorkunov B. Theoretical aspects of excitation and reception of acoustic waves for non-destructive testing of electromechanical and conversion systems. *2023 IEEE 4th KhPI Week on Advanced Technology (KhPIWeek),* IEEE, 2023, pp. 1–5, doi: 10.1109/KhPIWeek61412.2023.10312998.
7. Zaitsev R., Kirichenko M., Zaitseva L., Radoguz S. Automation measurement system of semiconductor devices parameters. *2020 IEEE 4th International Conference on Intelligent Energy and Power Systems (IEPS 2020)*, IEEE, 2020, pp. 126–129, doi:10.1109/IEPS51250.2020.9263136.
8. Medvedyk A., Lohoida M., Rybchak Z., Kulyna O. IT Slang: Development of Telegram Chatbot. *COLINS*, 2023, vol. 2, no. 3, pp. 152–162.
9. Levytskyi O. L. Rozrobka telehram botu dlia upravlinnia kontentom reklamnoi ahentsii: dyplom bakalavra. DUT, Kyiv, 2021. 44 p. Available at: https://duikt.edu.ua/repozitorii/ipz/2021/PPZ-52/Levytskyi Oleh.pdf (Accessed date: 23.02.2025).
10. Deng X., Yu Z. A meta-analysis and systematic review of the effect of chatbot technology use in sustainable education. *Sustainability*, 2023, vol. 15, no. 4, pp. 2940, doi:10.3390/su15042940.
11. Luo B., et al. A critical review of state-of-the-art chatbot designs and applications. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2022, vol. 12, no. 1, pp. e1434, doi:10.1002/widm.1434.
12. Silva M. I., et al. Review of conventional and advanced non-destructive testing techniques for detection and characterization of small-scale defects. *Progress in Materials Science*, 2023, vol. 138, pp. 101155, doi:10.1016/j.pmatsci.2023.101155.
13. Shi P., Su S., Chen Z. Overview of researches on the nondestructive testing method of metal magnetic memory: Status and challenges. *Journal of Nondestructive Evaluation*, 2020, vol. 39, art. 43, doi:10.1007/s10921-020-00688-z.

Відомості про авторів (About authors)

Зайцева Лілія Василівна – кандидат технічних наук, доцент, доцент кафедри комп'ютерних та радіоелектронних системи контролю та діагностики; Національний технічний університет «Харківський політехнічний інститут»; м. Харків, Україна; ORCID: 0000-0003-4405-1531; e-mail: Liliia.zaitseva@khpi.edu.ua.

Zaitseva Liliia – Candidate of Technical Sciences (Ph. D.), Docent, Associate Professor, Department of Computer and Radio-electronic Systems for Testing and Diagnostics, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; ORCID: 0000-0003-4405-1531; e-mail: Liliia.zaitseva@khpi.edu.ua.

Макаров Данило Олександрович – студент, кафедри комп'ютерних та радіоелектронних системи контролю та діагностики, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна; ORCID: 0009-0006-9928-7284; e-mail: danylo.makarov@cit.khpi.edu.ua.

Makarov Danylo student, Department of of Computer and Radio-electronic Systems for Testing and Diagnostics, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; ORCID: 0009-0006-9928-7284; e-mail: danylo.makarov@cit.khpi.edu.ua.

Будь ласка, посилайтесь на цю статтю наступним чином:

Зайцева Л. В., Макаров Д. О. Система збору статистичної інформації про дефекти за допомогою Telegram Bot: кроки та результати. *Вісник Національного технічного університету «ХПІ». Серія: Нові рішення в сучасних технологіях*. – Харків: НТУ «ХПІ». 2025. № 1 (23). С. 40-47. doi:10.20998/2413-4295.2025.01.05.

Please cite this article as:

Zaitseva L., Makarov D. A system for collecting statistical information about defects using Telegram Bot: steps and results. *Bulletin of the National Technical University "KhPI". Series: New solutions in modern technology*. – Kharkiv: NTU "KhPI", 2025, no. 1(23), pp. 40–47, doi:10.20998/2413-4295.2025.01.05.

Надійшла (received) 03.12.2024
Прийнята (accepted) 27.02.2025