

DOI: <https://doi.org/10.33216/1998-7927-2025-295-9-18-30>

УДК 004.89 : 004.738.5 : 658.8

НАПРЯМИ ОПТИМІЗАЦІЇ ПРОЦЕСІВ УПРАВЛІННЯ ІТ-ПОСЛУГАМИ

Погорєловська І.Д., Бур'ян С.С.

DIRECTIONS FOR OPTIMIZING IT SERVICE MANAGEMENT PROCESSES

Pogorielovska I.D., Burian S.S.

Розглянуто стандарти, методології та практики служб технічної підтримки. Проаналізовано проблеми ефективності традиційних каналів технічної підтримки у сучасних компаніях, запропоновано інноваційне гібридне рішення на основі інтеграції Telegram Bot з Jira Service Desk. Висвітлено архітектурні рішення, алгоритми роботи системи та особливості реалізації онлайн-взаємодії між користувачами та спеціалістами служби підтримки. Розроблено триступеневу архітектуру безпеки з розділенням на зони Інтернет, демілітаризовану зону та внутрішню корпоративну мережу. Запропоновано алгоритми авторизації, розподілу заявок та обробки онлайн-сесій з урахуванням навантаження спеціалістів та багатокритеріального підходу. Спроектована база даних з мінімальним набором інформації без конфіденційних бізнес-даних, що відповідає сучасним принципам інформаційної безпеки, та забезпечує швидкий доступ до необхідної інформації. Використано механізми кешування для зменшення навантаження на внутрішні системи та підвищення загальної продуктивності рішення. Запропонований механізм онлайн-консультацій перетворює традиційну асинхронну систему управління заявками в інструмент оперативної взаємодії в режимі реального часу, що критично важливо в умовах віддаленої роботи. Можливість миттєво зв'язатися зі спеціалістом та вирішити проблему в режимі живого діалогу значно підвищує ефективність технічної підтримки та задоволеність користувачів. Практична цінність розробленої системи полягає в можливості її впровадження в організаціях різного розміру та профілю діяльності. Розроблена система є масштабованою, безпечною та забезпечує всі необхідні можливості для ефективної організації технічної підтримки користувачів. Особливо актуальним це рішення є для українських компаній в контексті сучасних реалій

віддаленої роботи та необхідності забезпечення безперервності бізнес-процесів.

Ключові слова: управління ІТ-послугами, інтеграція систем, архітектура програмного забезпечення, Telegram Bot API, Jira Service Desk Telegram Bot API, Jira Service Desk

Вступ. В умовах інтенсивної цифровізації бізнес-процесів якісне управління ІТ-послугами в компанії відіграє все більш важливе значення і стає стратегічним активом. Основне завдання такого управління – надання якісних сервісів для збереження стійкості та розвитку бізнесу з використанням ІТ.

Оптимальне управління ІТ-послугами (ITSM) досягається, серед іншого, за рахунок вибору відповідних визначень процесів, стандартів процесів та програмного забезпечення, залежно від рівня формалізованості та регламентованості бізнес-процесів компанії. Водночас, в умовах інтенсивного розвитку ринкових відносин та зростання інновацій у галузі інформаційних технологій, управління ІТ-послугами як циклічна і безперервна діяльність, потребує постійного пошуку оптимальних підходів та засобів ефективного управління, що і зумовлює актуальність даного дослідження.

Аналіз останніх досліджень і публікацій. Про існування комплексу невирішених проблем, пов'язаних з технічною підтримкою користувачів корпоративних інформаційних систем свідчать результати досліджень, що висвітлені в роботах як вітчизняних так і зарубіжних вчених. Зокрема, у роботі [1, с.63] висвітлені проблеми, які пов'язані з

функціонуванням служби підтримки у закладах освіти, що зумовлені як складною інфраструктурою закладів так і непрогнозованою завантаженістю їхньої служби підтримки. Автори наголошують на необхідності запровадження систем HelpDesk, вказують на необхідність “зручного та зрозумілого для користувачів механізму, що дозволить направляти запити на сайт”, але не уточнюють, який саме спосіб взаємодії з цими системами є найкращим.

У дослідженні [2], також наголошено, що стан IT-інфраструктури у багатьох університетах є незадовільним, оскільки з її допомогою неможливо надавати якісні IT-послуги, що відповідають сучасним вимогам, і з гарантованою якістю, відповідно до стандартів та рекомендацій у галузі управління IT (ITIL, MOF, COBIT тощо).

Про необхідність безперервного, постійного удосконалення процесу технічної підтримки користувачів інформаційних систем банку зазначено у роботі [3]. Важливість *технологічної підтримки інформаційної інфраструктури Міністерства оборони України обґрунтована у роботі [4].*

Водночас, проведений аналіз свідчить, що існує достатньо велика кількість стандартів, методологій та практик організації ефективної роботи служб технічної підтримки, зокрема:

- ITIL (Information Technology Infrastructure Library) - комплексний набір практик і вказівок, які спрямовані на підвищення якості та ефективності IT-послуг в організаціях, узгодження IT-послуг з бізнес-стратегією організації [5]. Це гнучкий фреймворк, що охоплює процеси планування, надання, підтримки та постійного вдосконалення IT-послуг. Саме ITIL визначає технічну підтримку не як допоміжний процес, а як важливий елемент бізнесу.

- ISO/IEC 20000 - стандарт, що визначає вимоги до планування, створення, впровадження, експлуатації, моніторингу, перегляду та вдосконалення систем керування послугами [6]. Варто зазначити, що стандартизоване управління IT-сервісами є підґрунтям забезпечення якісних IT-послуг.

- COBIT (Control Objectives for Information and Related Technology) - рамкова система для гнучкого управління корпоративними IT, узгодження IT-стратегій з бізнес-цілями, інтеграції з іншими стандартами, такими як ISO та ITIL [7].

У системному огляді [8] зазначено, що більшість організацій започатковують ITSM на основі ITIL, проте існують й інші фреймворки, такі як Microsoft MOF, IBM ITPM та HP ITSM. Методологія MOF (Microsoft Operations Framework) є результатом адаптації ITIL до власних технологій компанії Microsoft [9]. Ця сукупність принципів та моделей спрямована на вдосконалення процесів керування операціями створення та підтримки надійних і економічно ефективних IT-послуг на базі продуктів Microsoft протягом усього їхнього життєвого циклу. IBM iTPM (Information Technology Portfolio Management) — це стратегія та набір інструментів від IBM, що орієнтовані на ефективне управління IT-портфелем компанії шляхом оптимізації витрат, підвищення цінності бізнесу та зниження ризиків [10]. Така стратегія допомагає організаціям узгоджувати свої технологічні інвестиції з бізнес-цілями, забезпечуючи прозорість та контроль над усіма IT-активами та проектами.

Також популярними в системах ITSM є **фреймворки:**

- CMMI (Capability Maturity Model Integration) - інтегрована модель, що визначає структурований підхід до управління та оцінювання процесів [11]. Використання цієї моделі дозволяє досягати вищих рівнів зрілості, підвищуючи ефективність результатів та знижуючи ризики, що, у свою чергу, сприяє розробці високоякісних продуктів та послуг.

- Six Sigma – методологія, що забезпечує структурований та систематичний спосіб виявлення, аналізу та усунення дефектів процесів, покращення якості, ефективності та задоволеності клієнтів (3,4 дефекти на мільйон можливостей) своїх послуг і продуктів, знизити витрати та підвищити задоволеність клієнтів [12]. Використовуючи методології та інструменти Six Sigma IT-компанії можуть покращити якість ITSM.

- eTOM (Enhanced Telecom Operations Map) – розширена карта телекомунікаційних операцій, що визначає загальну структуру телекомунікаційних бізнес-процесів, яка охоплює всі функції, необхідні для ефективного ведення телекомунікаційного бізнесу. Вона гарантує, що різні відділи та команди в телекомунікаційній організації узгоджені щодо своїх цілей, процесів та завдань, сприяючи спільному підходу до надання послуг [13]. Саме ця модель бізнес-процесів розглядається як важлива складова IT-інфраструктури Індустрії 4.0 у роботі [14, с. 88].

Про доцільність застосування фреймворку TOGAF для побудови корпоративної IT-архітектури, що узгоджується з бізнес-стратегією, та мови моделювання ArchiMate, яка дозволяє візуалізувати і синхронізувати всі складові цієї архітектури зазначено у роботі [15]. Там же підкреслено, що разом ці інструменти сприяють усуненню неузгодженостей, підвищенню ефективності та забезпеченню безперервного розвитку IT-архітектури.

Отже, структурований і методологічний виклад змісту розглянутих документів може бути основою для організації та забезпечення ефективного функціонування IT-відділів, підвищення їх здатності надавати якісні послуги, що сприятиме оперативному вирішенню проблем корпоративних користувачів, стабільному розвитку IT-інфраструктури та, у цілому, досягненню визначених бізнес-цілей компаній.

Крім того, є достатньо детальні огляди характеристик HelpDesk систем, які допомагають надавати ефективну, масштабовану та якісну підтримку користувачам корпоративних інформаційних систем, наприклад [16]. З різноманіття інструментів для підтримки взаємодії з клієнтами, компанії можуть обирати рішення, що відповідають їхнім потребам — платформи для онлайн спілкування, системи управління заявками, бази знань, підтримка в соціальних мережах та інструменти автоматизації.

Однак, на практиці, такі системи не завжди відповідають очікуванням через складність інтерфейсів, повільність обробки звернень та фрагментацію каналів комунікації. Крім того, більшість HelpDesk систем вимагають від користувача реєстрації на спеціальному порталі, запам'ятовування логін-пароля, заповнення складних форм з безліччю обов'язкових полів. Це створює певний психологічний бар'єр, оскільки простіше підійти до колеги з IT-відділу або навіть змиритися з ситуацією. Крім того, електронна пошта, як засіб комунікації, є "важким інтерфейсом", так як створює надлишкову інформацію в системі [17] і катастрофічно повільною для критичних ситуацій. Телефонна підтримка здається швидшим рішенням, але не залишає задокументованої історії. Корпоративні чати є зручним, але в них відсутні формалізація процесів, контроль виконання, аналітичні можливості [18].

Отже, результати аналізу сучасних досліджень і публікацій та власний досвід підтверджують, що для ефективного управління IT-послугами, служба підтримки, згідно з вимогами ITIL, повинна бути єдиною точкою контакту для запитів на обслуговування та вирішення інцидентів. Тому важливо мати надійні, сучасні засоби комунікації з користувачами, які зможуть забезпечити швидкий та зручний спосіб подачі звернення, його автоматичну фіксацію, структурування та передачу фахівцю.

Метою роботи є обґрунтування необхідності та розробка архітектури інтегрованої системи технічної підтримки, яка поєднує зручність популярного месенджера Telegram з потужністю професійної системи управління заявками Jira Service Desk, що забезпечить можливість як оперативного спілкування між користувачами та спеціалістами, так і оперативної обробки заявок на обслуговування.

Виклад основного матеріалу. Одним із варіантів вирішення проблеми ефективного управління IT-послугами є інтегрована система технічної підтримки, в якій усунено основний недолік тикет-систем — незручність для користувачів, але збережені при цьому всі переваги формалізованих процесів.

Ключовою ідеєю такого рішення стала розробка Telegram-бота як фронтенду для Jira Service Desk, де користувач працює, виключно, у звичному месенджері, а вся бізнес-логіка та збереження даних відбуваються у професійній системі.

Методологія ITIL, яка є міжнародним стандартом управління IT-сервісами, підкреслює важливість надання зручних каналів комунікації для користувачів при збереженні контролю та документування всіх звернень [5]. Саме досягнення балансу між зручністю та контролем було метою розробки інтегрованої системи.

Технічні можливості платформ для інтеграції. Telegram з його понад 700 мільйонами активних користувачів по всьому світу є ідеальною платформою для інтеграції. На відміну від корпоративних месенджерів, Telegram вже встановлений на пристроях більшості користувачів, має відкрите програмне середовище для розробників та підтримує розширені можливості для створення автоматизованих ботів. В Україні Telegram є одним з найпопулярніших месенджерів, що зумовлює його вибір для корпоративних рішень.

Серед опитаних українців, 81% користуються Telegram для спілкування, а 73% – для отримання новин [19].

Telegram Bot API надає потужний інструментарій для створення автоматизованих систем взаємодії з користувачами. Всі запити до серверів Telegram здійснюються через захищене з'єднання, що гарантує конфіденційність даних [20]. Основні можливості, які можна використовувати для системи технічної підтримки, включають: отримання повідомлень від користувачів, відправку текстових та медіафайлів, створення інтерактивних кнопок для швидкого вибору, редагування раніше відправлених повідомлень. Telegram також підтримує завантаження файлів розміром до 50 мегабайт, що достатньо для більшості випадків технічної підтримки – користувачі можуть надсилати скріншоти, логи помилок та інші документи безпосередньо через бот [21].

Jira Service Management є однією з найпотужніших платформ для управління IT-сервісами. Система пропонує розвинені можливості для програмної інтеграції через спеціальний програмний інтерфейс, який дозволяє зовнішнім додаткам створювати заявки, отримувати інформацію про їх статус, додавати коментарі та виконувати інші операції. Типовий процес обробки заявки технічної підтримки включає такі етапи: нова заявка, взята в роботу, очікування відповіді від користувача,

очікування дій від спеціалістів, проблема вирішена, заявка закрита. Кожен перехід між етапами можна налаштувати з урахуванням специфіки компанії, додавати автоматичні перевірки та дії. [22]. Гнучка система налаштування робочих процесів є однією з найсильніших сторін Jira, що дає змогу адаптувати систему під свої специфічні потреби та інтегрувати її з іншими корпоративними системами.

Розроблена архітектура системи. У процесі дослідження була розроблена триступенева архітектура системи з розділенням на зони відповідно до принципів інформаційної безпеки (рис.1). Така архітектура забезпечує баланс між зручністю використання, продуктивністю та безпекою даних, що особливо важливо в контексті сучасних кіберзагроз, які детально висвітлені у роботі [23, с. 841-865].

Перший рівень – це зона Інтернет, в якій знаходяться кінцеві користувачі з додатками Telegram на своїх пристроях. Ця зона повністю ізолювана від внутрішніх компонент системи і має доступ лише до другого рівня через зашифроване з'єднання. Навіть якщо пристрій користувача буде зламано, зломисник не зможе отримати прямий доступ до внутрішніх систем компанії.

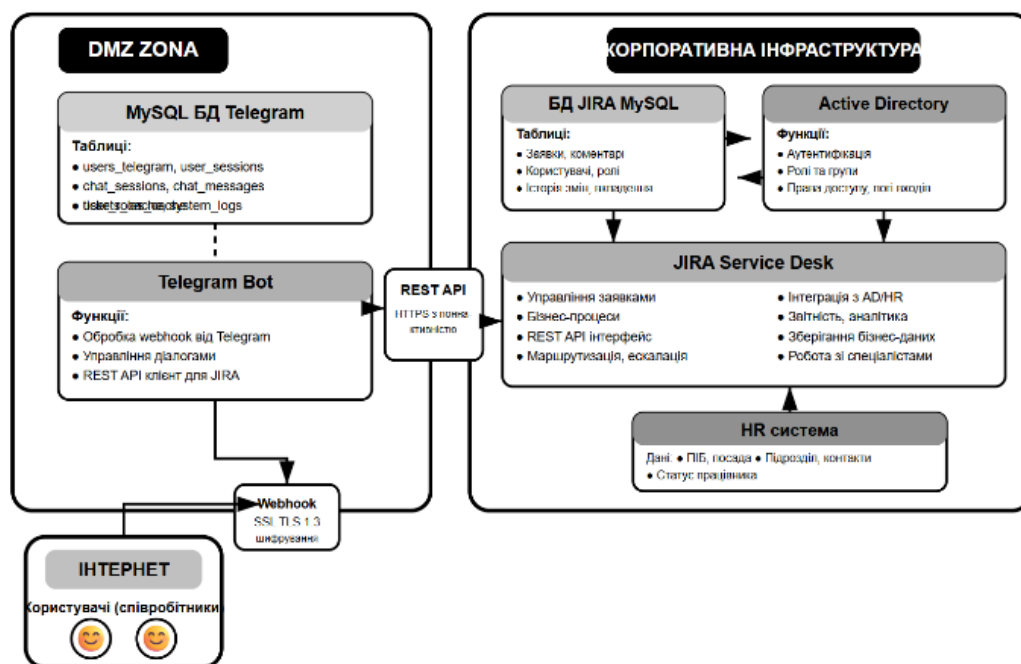


Рис. 1. Триступенева архітектура системи з розділенням на зони відповідно до принципів інформаційної безпеки (розроблено автором)

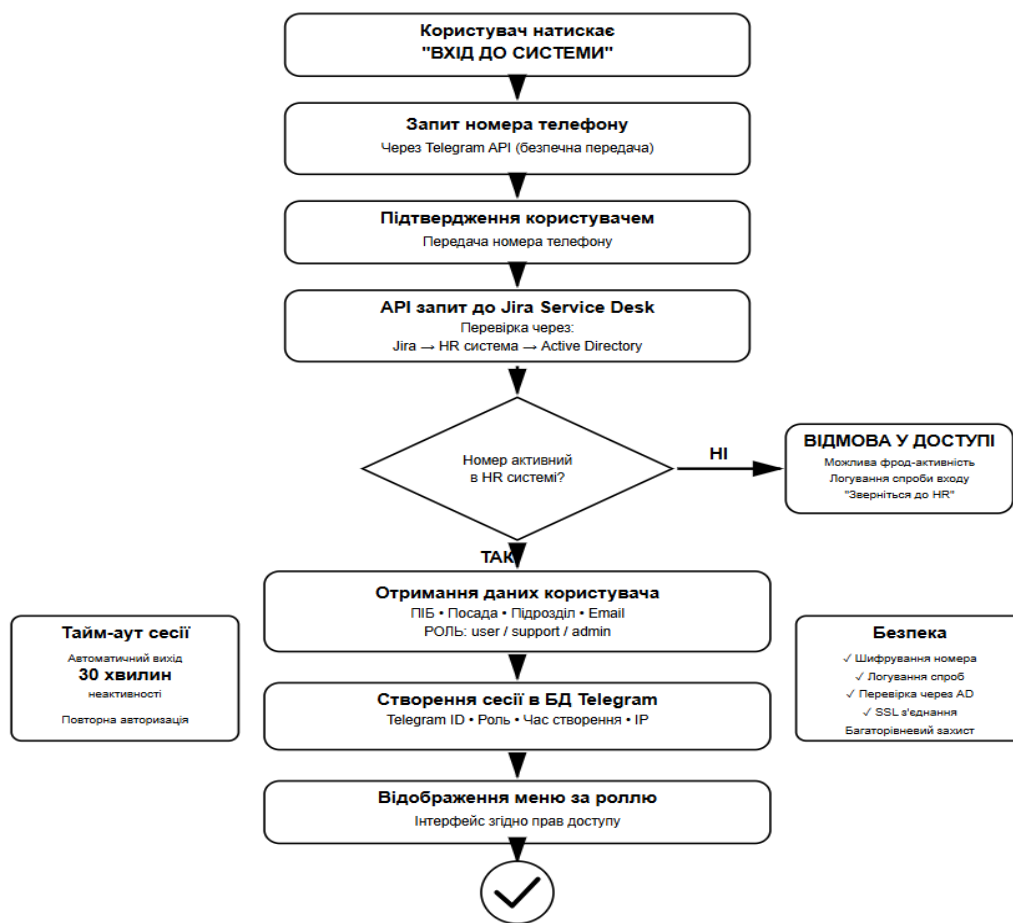


Рис. 2. Алгоритм авторизації користувачів в системі (розроблено автором)

Другий рівень – демілітаризована зона, яка є буферною зоною між Інтернетом та внутрішньою мережею. Тут розміщені два ключові компоненти: серверний додаток Telegram Bot, який обробляє повідомлення користувачів та управляє інтерфейсом, та база даних для зберігання тимчасової інформації – активних сесій користувачів, історії онлайн-чатів, системних логів. Важливо підкреслити, що конфіденційна бізнес-інформація у цій зоні не зберігається.

Третій рівень – внутрішня корпоративна мережа, де розміщені критичні системи організації: Jira Service Desk з базою даних заявок, система управління користувачами Active Directory, HR система з інформацією про співробітників. Telegram Bot не має прямого доступу до цих систем – вся взаємодія відбувається виключно через програмний інтерфейс Jira, що є єдиною контрольованою точкою входу.

Типовий процес обробки запиту користувача виглядає наступним чином. Користувач надсилає повідомлення в Telegram, яке через зашифроване з'єднання надходить до бота в демілітаризованій зоні. Бот перевіряє, чи активна сесія користувача, формує запит до Jira у внутрішній мережі. Jira виконує необхідні дії, звертаючись при потребі до системи управління користувачами або HR системи, повертає результат боту, який формує зрозумілу відповідь користувачу.

Алгоритм авторизації користувачів. Розроблений алгоритм авторизації забезпечує надійну ідентифікацію користувачів на основі їхнього номера телефону. При першому запуску бота користувач бачить привітальне повідомлення та кнопку для початку процесу входу в систему (рис. 2). Бот запитує номер телефону через вбудований механізм Telegram, який дозволяє користувачу поділитися своїм номером одним натисканням без необхідності вводити його вручну. Це не лише зручно для

користувача, але й безпечно, оскільки виключена можливість помилки при введенні або спроби використати чужий номер телефону.

Після отримання номера бот надсилає запит до HR системи через програмний інтерфейс Jira для перевірки, чи є цей номер телефону в корпоративній базі співробітників. Якщо номер не знайдено, система відмовляє в доступі та обов'язково фіксує цю спробу в журналі безпеки для можливого розслідування. Якщо ж номер підтверджено, система отримує повну інформацію про користувача: повне ім'я, посаду, відділ, роль в системі технічної підтримки.

Після успішної верифікації створюється сесія користувача, яка зберігає інформацію про його ідентифікатор в Telegram, присвоєну роль, час початку сесії та час останньої активності. Система відображає меню відповідно до ролі користувача – звичайні користувачі бачать можливість створити заявку або переглянути

свої заявки, спеціалісти підтримки отримують доступ до списку призначених їм заявок та можливості приймати запити на онлайн-консультації.

Критично важливим елементом безпеки є автоматичне завершення сесії через тридцять хвилин неактивності. При кожній дії користувача система перевіряє, коли була остання активність, і якщо минуло більше встановленого часу, вимагає повторної авторизації. Це запобігає несанкціонованому доступу у випадку, якщо користувач залишив пристрій без нагляду або втратив його.

Алгоритм розподілу заявок між спеціалістами. Ефективний розподіл заявок є ключовим фактором продуктивності служби підтримки. Тому було розроблено алгоритм, який забезпечує справедливий та оптимальний розподіл навантаження між всіма доступними спеціалістами (рис. 3).

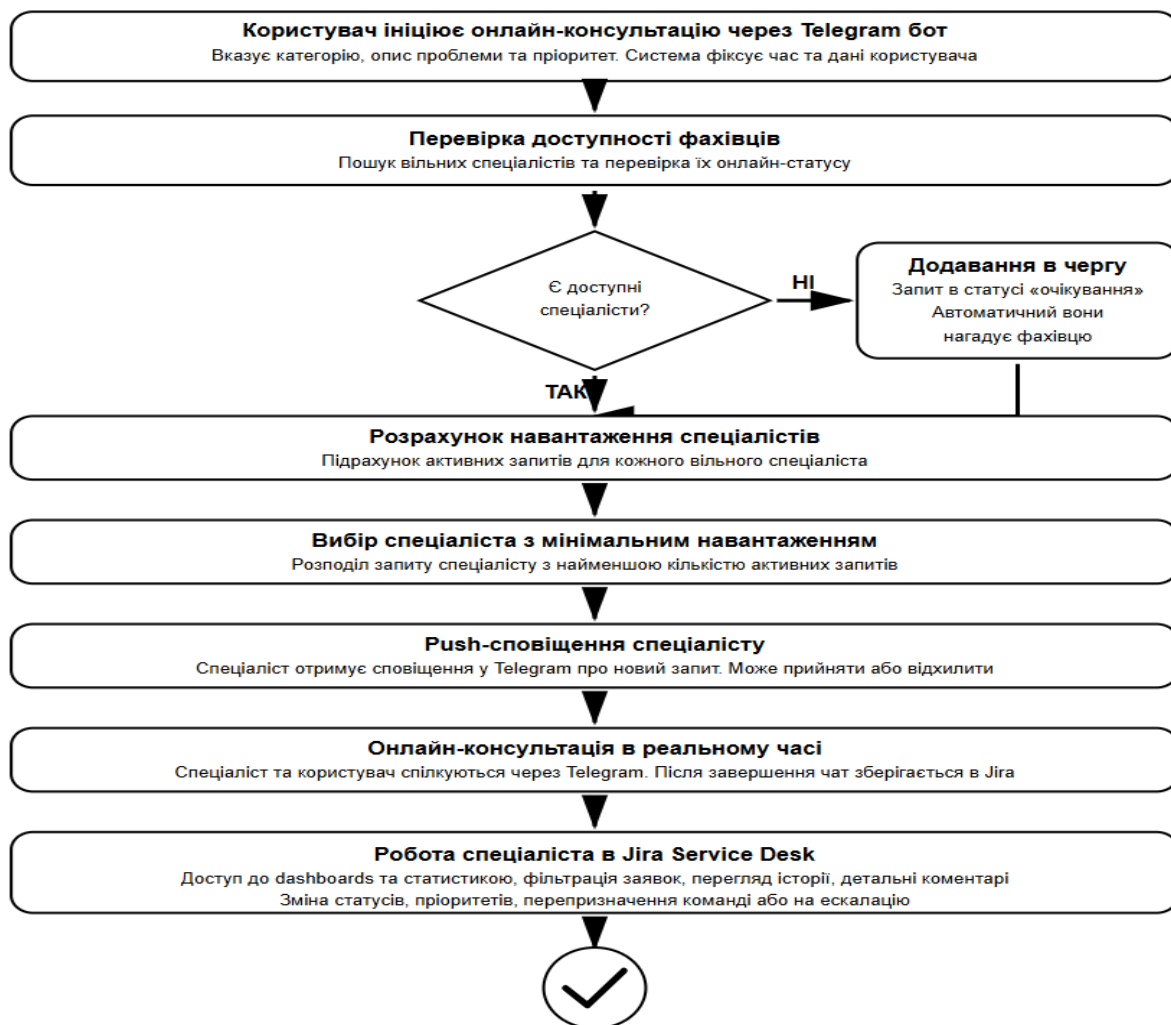


Рис. 3. Алгоритм розподілу заявок між спеціалістами служби підтримки (розроблено автором)

У разі, коли користувач створює заявку через Telegram Bot, він самостійно заповнює всю необхідну інформацію: обирає тип проблеми з підготовленого переліку категорій, встановлює пріоритет заявки (критичний, високий, середній, низький), вказує детальний опис проблеми, може прикріпити скріншоти або інші файли для кращого розуміння ситуації. Після заповнення всіх полів система автоматично створює заявку в Jira Service Desk через програмний інтерфейс та присвоює їй унікальний номер для відстеження.

Одразу після створення заявки система запускає процес автоматичного розподілу. Важливо підкреслити, що всі спеціалісти технічної підтримки приймають в роботу проблеми будь якої складності, а вже за потреби вирішення проблеми на вищому рівні, передають колегам вищої компетенції – система не враховує спеціалізацію чи компетенції при розподілі заявок.

Алгоритм розподілу базується на трьох основних показниках для кожного спеціаліста. По-перше, це поточне навантаження – кількість активних заявок, які зараз знаходяться в роботі у спеціаліста в статусах "нова", "в роботі" та "очікує відповіді користувача". По-друге, система враховує час закриття останньої заявки – це дозволяє більш рівномірно розподіляти нові звернення між спеціалістами, які тільки завершили роботу над попередньою задачею. По-третє, враховується загальна кількість закритих заявок за поточний період – це забезпечує баланс навантаження в довгостроковій перспективі.

Система розраховує інтегральний показник доступності для кожного спеціаліста, який комбінує всі три метрики. Заявка автоматично призначається спеціалісту з найкращим показником доступності – тобто тому, хто має найменше активних заявок, давно завершив попередню роботу та має меншу загальну кількість закритих заявок за період. Обраний спеціаліст миттєво отримує сповіщення в Telegram з базовою інформацією про нову заявку: номер, пріоритет, короткий опис проблеми та можливість відразу перейти до детального перегляду в Jira.

При розробці алгоритму навантаження на спеціалістів служби технічної підтримки був врахований збалансований підхід, про важливість якого наголошено у праці [24]. Саме цей підхід дозволяє системі забезпечувати справедливий розподіл навантаження між всіма членами команди технічної підтримки.

Механізм онлайн-консультацій. Найбільш інноваційною частиною розробленої системи є механізм онлайн-консультацій у реальному часі. На відміну від традиційного підходу систем управління заявками, де вся комунікація відбувається асинхронно через коментарі до заявки, цей механізм дозволяє користувачу миттєво зв'язатися зі спеціалістом для оперативного вирішення проблеми (див. рис. 3). Процес починається з того, що користувач створює заявку через Telegram Bot, заповнюючи всю необхідну інформацію: тип проблеми, пріоритет, детальний опис. Після створення заявки користувач може натиснути кнопку "Зв'язатися зі спеціалістом онлайн", якщо проблема потребує термінового вирішення або детального обговорення в режимі діалогу. У разі, коли користувач ініціює запит на онлайн-консультацію, система перевіряє доступність спеціалістів технічної підтримки. Якщо є вільні спеціалісти, система запускає той самий алгоритм розподілу, що використовується для звичайних заявок. Система аналізує три ключові показники для кожного доступного спеціаліста: поточне навантаження активними заявками, час закриття останньої заявки та загальну кількість закритих заявок за період.

На основі цих показників система розраховує інтегральний коефіцієнт навантаження та обирає оптимального спеціаліста – того, хто має найменше поточних задач, нещодавно завершив попередню роботу і має загалом менше закритих заявок. Обраний спеціаліст отримує сповіщення в Telegram про запит на онлайн-консультацію з інформацією про заявку та користувача. Він може прийняти запит одним натисканням кнопки, після чого встановлюється прямий зв'язок між користувачем та спеціалістом через бот.

Особливо важливою є ситуація, коли всі спеціалісти зайняті іншими консультаціями або активними заявками. У такому випадку запит на онлайн-консультацію автоматично переходить у статус очікування, і користувач отримує відповідне повідомлення з приблизним часом очікування. Система фіксує час створення запиту та його пріоритет для подальшої обробки.

Коли один зі спеціалістів завершує роботу над поточною заявкою або онлайн-консультацією, система автоматично перевіряє наявність запитів у черзі очікування. Якщо такі запити є, система запускає алгоритм розподілу серед всіх спеціалістів, які щойно звільнилися або мають найменше навантаження. Запити з

черги обробляються з урахуванням їхнього пріоритету та часу очікування – критичні та більш старі запити отримують вищий пріоритет при розподілі.

З моменту встановлення зв'язку всі повідомлення, які користувач надсилає боту, автоматично передаються призначеному спеціалісту, і навпаки – відповіді спеціаліста миттєво доставляються користувачу. Система створює окрему сесію чату, яка зберігає інформацію про обох учасників розмови та номер відповідної заявки в Jira.

Всі повідомлення в онлайн-чаті подвійно зберігаються: тимчасово в базі даних Telegram, для швидкого відображення, та паралельно додаються як коментарі до заявки в Jira через програмний інтерфейс. Це забезпечує повну документацію спілкування для подальшого аналізу та звітності. Сесія онлайн-консультації завершується або вручну, коли будь-який з учасників натискає кнопку завершення розмови, або автоматично після десяти хвилин без повідомлень.

Після завершення консультації вся історія чату остаточно синхронізується з Jira, статус спеціаліста змінюється на "вільний", і система автоматично перевіряє наявність нових запитів у черзі очікування. Користувач може оцінити якість консультації за простою шкалою від одного до п'яти балів, що допомагає відстежувати ефективність роботи команди підтримки.

Цей механізм принципово змінює підхід до технічної підтримки, перетворюючи традиційну систему управління заявками в інструмент для оперативної взаємодії. Це критично важливо в сучасних умовах, особливо коли спеціалісти можуть працювати віддалено з різних локацій. Онлайн-консультації дозволяють значно скоротити час вирішення проблем, покращити якість діагностики завдяки можливості ставити уточнюючі питання в режимі реального часу, та підвищити загальну задоволеність користувачів якістю обслуговування.

Організація бази даних. Для забезпечення швидкодії та надійності функціонування системи була спроектована база даних, яка зберігає необхідну інформацію для роботи Telegram Bot. Ключовим принципом проектування було збереження мінімального набору даних без конфіденційної бізнес-інформації, яка залишається виключно у внутрішній корпоративній мережі.

База даних складається з семи основних таблиць, які забезпечують різні аспекти функціонування системи (рис. 4). Таблиця **USERS_TELEGRAM** призначена для зберігання інформації про користувачів: їхній унікальний ідентифікатор в Telegram, зашифрований номер телефону для додаткової безпеки, посилання на профіль користувача в Jira, присвоєну роль в системі, дату створення акаунту та час останнього входу.

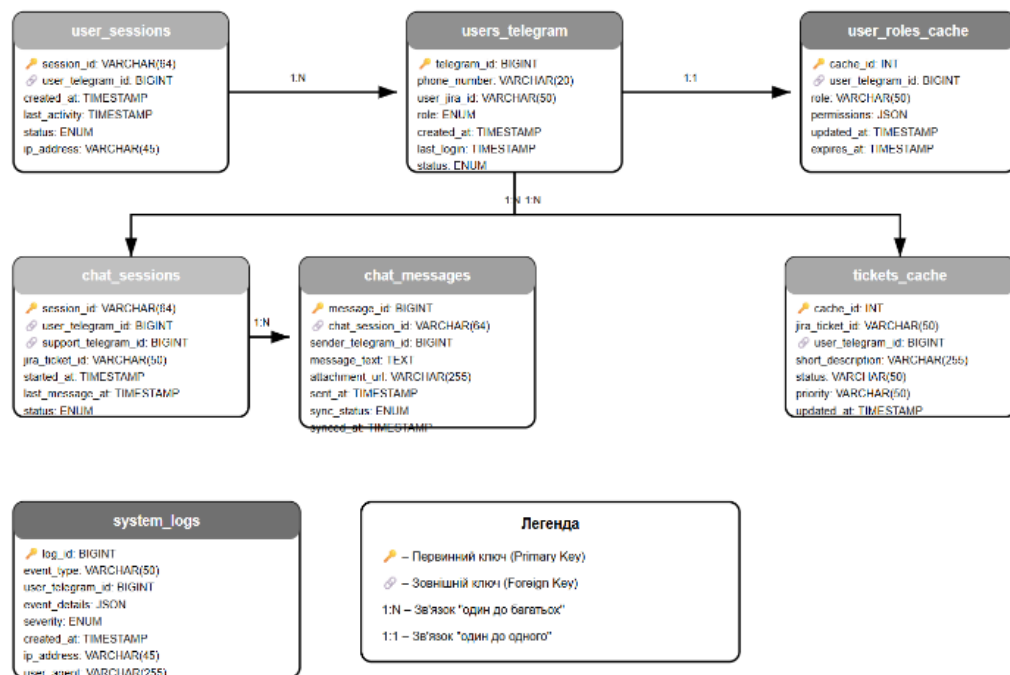


Рис. 4. Схема бази даних системи (розроблено автором)

Таблиця `USER_SESSIONS` керує активними сесіями користувачів, зберігаючи час створення сесії, час останньої активності та поточний статус. Саме ця таблиця дозволяє реалізувати механізм автоматичного завершення сесій після періоду неактивності. Для оптимізації продуктивності та зменшення кількості запитів до внутрішніх систем організації, система використовує таблицю кешу `USER_ROLES_CACHE`, яка тимчасово зберігає інформацію про ролі та дозволи користувачів.

Для забезпечення роботи механізму онлайн-консультацій передбачено дві спеціалізовані таблиці. Таблиця `CHAT_SESSIONS` зберігає інформацію про активні сесії чатів: учасників розмови, номер відповідної заявки, час початку та статус сесії. Таблиця `CHAT_MESSAGES` містить історію всіх повідомлень в онлайн-чатах з інформацією про відправника, текст повідомлення, вкладені файли та статус синхронізації з Jira.

Додатково база даних містить таблицю кешу заявок `TICKETS_CACHE`, яка дозволяє швидко відображати список заявок користувача без постійних звернень до Jira. У ній зберігається лише базова інформація: номер заявки, короткий опис проблеми, поточний статус, пріоритет та час створення.

Таблиця `SYSTEM_LOG` – це системний журнал, у якому фіксуються всі важливі події, що відбуваються в системі: спроби авторизації, створення заявок, помилки при роботі з програмними інтерфейсами, початок та завершення онлайн-чатів.

Важливо підкреслити, що розроблена база даних є лише допоміжним сховищем для забезпечення швидкої роботи бота. Всі конфіденційні бізнес-дані – детальні описи проблем, історія обробки заявок, коментарі спеціалістів, вкладені файли з можливою чутливою інформацією – зберігаються виключно у внутрішній корпоративній мережі в базі даних Jira Service Desk. Персональні дані в базі Telegram зберігаються в зашифрованому вигляді відповідно до вимог чинного законодавства про захист персональних даних.

При розробці бази даних були враховані вимоги, визначені у [25], щодо важливості правильного розділення даних між різними сховищами з урахуванням їхньої критичності та вимог до швидкості доступу. Саме такий підхід забезпечує баланс між продуктивністю системи та безпекою даних.

Протоколи взаємодії та забезпечення безпеки. Для взаємодії між компонентами системи використовуються сучасні протоколи та методи комунікації. Основою є REST підхід, сутність якого розкрито в дослідженнях [26] та, який забезпечує простоту та надійність обміну даними між Telegram Bot та Jira Service Desk.

Для забезпечення роботи в реальному часі система використовує механізм вебхуків – спеціальних повідомлень про події. Коли в Jira відбувається важлива подія – створення нової заявки, зміна статусу, додавання коментаря – система автоматично надсилає повідомлення боту. Аналогічно Telegram використовує вебхуки для миттєвої доставки повідомлень від користувачів до бота. Це дозволяє скоротити затримку з тридцяти-шістдесяти секунд при традиційному підході до одної-двох секунд.

Безпека забезпечується на декількох рівнях одночасно. Всі з'єднання між компонентами системи використовують сучасні протоколи шифрування, які гарантують конфіденційність та цілісність даних при передачі. Спеціальні ключі доступу до програмних інтерфейсів регулярно оновлюються для мінімізації ризиків компрометації. Система має вбудований механізм обмеження кількості запитів – максимум тридцять запитів на секунду від одного користувача, що захищає від атак типу "відмова в обслуговуванні".

Всі вхідні дані проходять обов'язкову перевірку та очищення для захисту від спроб впровадження шкідливого коду. Ведеться повний журнал аудиту всіх операцій в системі, що дозволяє відстежити будь-які підозрілі дії та провести розслідування інцидентів безпеки. Особливу увагу приділено захисту персональних даних відповідно до європейських стандартів GDPR та українського законодавства, що особливо важливо в умовах зростання кіберзагроз. Розроблена система враховує сучасні вимоги до безпеки на всіх рівнях архітектури.

Переваги інтегрованого підходу. Комплексний підхід до розробки архітектури інтегрованої системи технічної підтримки, про важливість застосування якого при розробці складних систем, наголошено у роботі [27], дозволив урахувати потреби всіх зацікавлених сторін. Створена система поєднала кращі властивості двох технологій – зручність популярного месенджера та потужність професійної системи управління заявками. Користувачі отримали можливість оперативно формувати та надсилати заявку, не виходячи з

месенджера, який вже встановлений на їхніх пристроях. При цьому не потрібно пам'ятати адреси корпоративних порталів, логіни та паролі до них, шукати потрібні розділи у складних меню.

Механізм онлайн-консультацій дозволяє користувачам отримати миттєву допомогу у критичних ситуаціях, не очікуючи на відповідь електронною поштою чи в черзі на телефонній лінії. Проблеми можуть вирішуватися в режимі живого діалогу, що особливо є цінним для нетехнічних користувачів, яким іноді складно детально сформулювати проблему в письмовій формі, - в розмові уточнюючі запитання спеціаліста дозволяють швидко з'ясувати суть проблеми.

Для спеціалістів технічної підтримки система забезпечує зручний інструмент оперативної комунікації через Telegram при збереженні всіх переваг професійної системи Jira. Вони отримують миттєві сповіщення про нові заявки прямо в месенджер, можуть швидко переглянути базову інформацію, і за потреби перейти до повнофункціонального інтерфейсу Jira. Історія всіх онлайн-консультацій автоматично зберігається в системі, що важливо для подальшого аналізу та формування висновків та звітів.

Для керівництва служби підтримки та адміністраторів системи основною перевагою є повна прозорість всіх процесів. Кожне звернення фіксується, кожна дія документується, накопичується статистика для аналізу. Відомо про кількість заявок, що обробляється щодня, які категорії проблем є найпоширенішими, як швидко спеціалісти реагують на звернення, наскільки користувачі задоволені якістю обслуговування. Ці дані дозволяють приймати обґрунтовані рішення щодо оптимізації служби підтримки, розподілу ресурсів та навчання персоналу.

У дослідженнях систем штучного інтелекту відзначають, що збір та аналіз великих обсягів структурованих даних про роботу служби підтримки створює основу для впровадження інтелектуальних систем в майбутньому [28]. Наприклад, система може навчитися автоматично пропонувати рішення типових проблем або передбачати потенційні проблеми до їх виникнення. Це визначає наступний етап розвитку розробленої інтегрованої системи.

Розроблена система має значний практичний потенціал для українських компаній різного розміру. Для малих та середніх компаній вона дозволяє організувати професійну технічну

підтримку без великих інвестицій в складну інфраструктуру. Telegram є безкоштовним месенджером, Jira Service Management має доступні тарифні плани для невеликих команд, а розробка самого бота не вимагає значних ресурсів завдяки добре документованому програмному інтерфейсу.

Особливо актуальною система є в контексті сучасних умов роботи в Україні. Віддалена робота стала нормою для багатьох компаній і традиційні методи технічної підтримки вже не працюють. Telegram Bot дозволяє організувати ефективну підтримку незалежно від того, де фізично знаходяться користувачі та спеціалісти.

Висновки. Результати проведеного дослідження свідчать про існування комплексу невирішених проблем, пов'язаних з технічною підтримкою користувачів корпоративних інформаційних систем. Водночас, запровадження в практику використання стандартів, методологій (ITIL, COBIT, ISO/IEC 20000 інш.) та сучасних систем автоматизації ITSM сприяє їх ефективному вирішенню.

Розроблена інтегрована система технічної підтримки на базі Telegram Bot та Jira Service Desk успішно поєднала зручність популярного месенджера з потужністю професійної систем управління заявками. Триступенева архітектура системи з розподілом на зони Інтернет, демілітаризовану зону та внутрішню корпоративну мережу забезпечує високий рівень інформаційної безпеки навіть при використанні публічного месенджера. Компрометація будь-якого рівня не надає доступу до критичних систем організації завдяки принципу ізоляції та єдиній контрольованій точці входу через програмний інтерфейс Jira.

Розроблені алгоритми авторизації на основі номера телефону з верифікацією через HR систему, автоматичного розподілу заявок з урахуванням навантаження спеціалістів та багатокритеріального підходу, обробки онлайн-сесій в реальному часі дозволяють значно оптимізувати процеси технічної підтримки. Система забезпечує баланс між швидкістю реагування та якістю документування всіх звернень.

Найбільш інноваційним аспектом розробленої системи є механізм онлайн-консультацій, який перетворює традиційну асинхронну систему управління заявками в інструмент оперативної взаємодії в режимі реального часу. Це критично важливо в сучасних умовах віддаленої роботи, коли

спеціалісти та користувачі можуть знаходитися в різних локаціях. Можливість миттєво зв'язатися зі спеціалістом та вирішити проблему в режимі живого діалогу значно підвищує ефективність технічної підтримки та задоволеність користувачів.

Спроектowana база даних з мінімальним набором інформації без конфіденційних бізнес-даних відповідає сучасним принципам інформаційної безпеки та забезпечує швидкий доступ до необхідної інформації. Використання механізмів кешування дозволяє зменшити навантаження на внутрішні системи організації та підвищити загальну продуктивність даного рішення.

Практична цінність розробленої системи полягає в можливості її впровадження в компаніях різного розміру та профілю діяльності. Розроблена система є масштабованою, безпечною та сприяє ефективній організації технічної підтримки користувачів.

Перспективні дослідження будуть спрямовані на розвиток та удосконалення системи HelpDesk за рахунок впровадження елементів штучного інтелекту для автоматичної обробки типових запитів, розробки модулів машинного навчання для прогнозування потенційних проблем, інтеграції з іншими популярними месенджерами (Viber, WhatsApp) для надання користувачам можливості вибору більш зручного каналу комунікації, впровадження аналітичних систем для виявлення закономірностей та подальшої оптимізації роботи служби технічної підтримки. Реалізація цих напрямів сприятиме, в цілому, стабільному, прогнозованому та ефективному функціонуванню корпоративної IT-інфраструктури компанії.

Л і т е р а т у р а

1. Круглик В. С., Плечій О. О. Концепція організації інформаційної системи технічної підтримки // Інформаційні технології в освіті. 2012. № 12. С. 61–66.
2. Василів В. Б. Практичні підходи до формування служби технічної підтримки користувачів інформаційної системи університету // Бізнес Інформ. 2018. №4. С. 371–377.
3. Заховалко Т. В., Максишко Н. К., Олешко О. В. Аналіз та удосконалення процесу технічної підтримки користувачів інформаційних систем банку // Вісник Запорізького Національного університету. 2014. №3(14). С. 64–76.
4. Дядечко А. О., Даценко І. П., Головченко О. В. Концептуальні аспекти технологічної підтримки інформаційної інфраструктури Міністерства оборони України // Сучасні інформаційні технології у сфері безпеки та оборони. 2024. № 3(51). С. 96–107.
5. ITIL® and IT Service Management. URL: <http://www.itil.org.uk/> (дата звернення 10.06.2025).
6. COBIT an ISACA Framework. URL: <https://www.isaca.org/resources/cobit#1> (дата звернення 10.06.2025).
7. ISO/IEC 20000 – міжнародний стандарт для управління та обслуговування IT сервісів. URL: <https://tms.ua/iso-iec-20000/> (дата звернення 10.06.2025).
8. Iden Jon, Eikebrokk Tom Roar. Implementing IT Service Management: A systematic literature review. International Journal of Information Management. Volume 33, Issue 3, June 2013, p. 512–523. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0268401213000078> (дата звернення 05.09.2025).
9. Microsoft Operations Framework 4.0. URL: [https://learn.microsoft.com/en-us/previous-versions/tn-archive/cc506049\(v=technet.10\)](https://learn.microsoft.com/en-us/previous-versions/tn-archive/cc506049(v=technet.10)) (дата звернення 10.06.2025).
10. IT Portfolio Management (ITPM) URL: [https://cio-wiki.org/wiki/IT_Portfolio_Management_\(ITPM\)](https://cio-wiki.org/wiki/IT_Portfolio_Management_(ITPM)) (дата звернення 10.06.2025).
11. CMMIs Guide to Bridging the Strategy Execution Gap. URL: <https://cmmiinstitute.com/resource-files/public/cmmis-guide-to-bridging-the-strategy-execution-gap> (дата звернення 10.06.2025).
12. Rodriguez Diego. Understanding Six Sigma in the IT Industry. URL: <https://www.invensislearning.com/blog/six-sigma-it-transformation/> (дата звернення 10.06.2025).
13. E TOM: Enhanced Telecom Operation Map. URL: <https://www.scribd.com/doc/50556336/CCU-eTOM> (дата звернення 10.06.2025).
14. Довгий С. О., Копійка О. В. IT-інфраструктура як базова складова цифрової трансформації. Монографія. К.: ТОВ «Видавництво «Юстон», 2023. 458 с.
15. Безпечна корпоративна архітектура: TOGAF та ArchiMate для сталого зростання. URL: <https://surl.lu/vnytcx> (дата звернення 10.06.2025).
16. Олешко Д. Повний гід з 25 найкращих програмних інструментів для обслуговування клієнтів. URL: <https://surl.li/spjmjb> (дата звернення 10.06.2025).
17. Bosu Michael, Abuaiadah Diab et al. Evaluation of IT Service Desk: A Case Study. URL: https://www.academia.edu/70472216/Evaluation_of_IT_Service_Desk_A_Case_Study (дата звернення 10.06.2025).
18. Бур'ян С. С., Погореловська І. Д. Інтегрована система служби технічної підтримки на основі Telegram і Jira Service Desk. Економіка і бізнес: нові реалії, технологічні тренди, цифрові

- інновації: збірник матеріалів Міжнародної науково-практичної конференції здобувачів вищої освіти та молодих вчених (Львів-Ірпін, 15 травня 2025 р.). Львів: ЛНУ ім. Івана Франка, 2025. С. 530-533.
19. Українські медіа, ставлення та довіра у 2024 р. Опитування USAID-Internews щодо споживання медіа. URL: https://drive.google.com/file/d/1kwsclr3Qm2QaqaIVv0_l4saWRFY_NXWb/view?pli=1 (дата звернення 05.09.2025).
 20. Telegram FAQ. URL: <https://telegram.org/faq> (дата звернення 23.07.2025).
 21. Telegram Bot API server. URL: <https://github.com/tldlib/telegram-bot-api> (дата звернення 23.07.2025).
 22. Atlassian Corporation. Jira REST API Documentation. URL: <https://developer.atlassian.com/cloud/jira/platform/> ((дата звернення 07.07.2025).
 23. OWASP Top 10 API Security Risks – 2023. URL: <https://owasp.org/API-Security/editions/2023/en/0x11-t10/> (дата звернення 05.09.2025).
 24. Волошин О. Ф. Моделі та методи прийняття рішень: навч. посіб. для студ. вищ. навч. закл. /О.Ф. Волошин, С.О. Мащенко. 2-ге вид., перероб. та допов. К.: Видавничополіграфічний центр "Київський університет", 2010. 336 с.
 25. Teorey T. J., Morgan T. Et al. Database Design: Know It All. (Morgan Kaufmann Know It All) : Morgan Kaufmann, 2008. 368 p.
 26. Fielding R. T., Taylor R. N. Principled design of the modern Web architecture. In Proceedings of the 2000 International Conference on Software Engineering (ICSE 2000), Limerick, Ireland, June 2000. P. 407–416.
 27. Згуровський М. З., Панкратова Н. Д. Основи системного аналізу. К.: ВПЦ "Київський університет", 2007. 544 с.
 28. Литвин В. В., Пасічник В. В., Яцишин А. В. Системи штучного інтелекту. Львів: "Магнолія 2006", 2017. 676 с.
 1. Kruglyk, V. S., Plechii, O. O. Kontseptsiia orhanizatsii informatsiinoi systemy tekhnichnoi pidtrymky // *Informatsiini tekhnolohii v osviti*, 2012. #12. S. 61–66.
 2. Vasyliiv, V. B. Praktychni pidkhody do formuvannia sluzhby tekhnichnoi pidtrymky korystuvachiv informatsiinoi systemy universytetu // *Biznes Inform*, 2018. #4. S. 371–377.
 3. Zakhovalko, T. V., Maksyshko, N. K., & Oleshko, O. V. Analiz ta udoskonalennia protsesu tekhnichnoi pidtrymky korystuvachiv informatsiinykh system banku // *Visnyk Zaporizkoho Natsionalnoho Universytetu*, 2014. #3(14). S. 64–76.
 4. Diadechko, A. O., Datsenko, I. P., Holovchenko, O. V. Kontseptualni aspekty tekhnolohichnoi pidtrymky informatsiinoi infrastruktury Ministerstva oborony Ukrainy // *Suchasni informatsiini tekhnolohii u sferi bezpeky ta oborony*, 2024. #3(51), S. 96–107.
 5. ITIL® and IT Service Management. URL: <http://www.itil.org.uk/> (Accessed: 10.06.2025).
 6. COBIT – an ISACA Framework. URL: <https://www.isaca.org/resources/cobit#1> (Accessed: 10.06.2025).
 7. ISO/IEC 20000 – mizhnarodnyi standart dlia upravlinnia ta obsluhovuvannia IT servisiv. URL: <https://tms.ua/iso-iec-20000/> (Accessed: 10.06.2025).
 8. Iden Jon, Eikebrokk Tom Roar. Implementing IT Service Management: A systematic literature review. International Journal of Information Management. Volume 33, Issue 3, June 2013, p. 512-523. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0268401213000078> (Accessed: 05.09.2025).
 9. Microsoft Operations Framework 4.0. URL: [https://learn.microsoft.com/en-us/previous-versions/tn-archive/cc506049\(v=technet.10\)](https://learn.microsoft.com/en-us/previous-versions/tn-archive/cc506049(v=technet.10)) (Accessed: 10.06.2025).
 10. IT Portfolio Management (ITPM). URL: [https://cio-wiki.org/wiki/IT_Portfolio_Management_\(ITPM\)](https://cio-wiki.org/wiki/IT_Portfolio_Management_(ITPM)) (Accessed: 10.06.2025).
 11. CMMI's Guide to Bridging the Strategy Execution Gap. URL: <https://cmmiinstitute.com/resource-files/public/cmmis-guide-to-bridging-the-strategy-execution-gap> (Accessed: 10.06.2025).
 12. Rodriguez Diego. Understanding Six Sigma in the IT Industry. URL: <https://www.invensislearning.com/blog/six-sigma-it-transformation/> (Accessed: 10.06.2025).
 13. E TOM: Enhanced Telecom Operation Map. URL: <https://www.scribd.com/doc/50556336/CCU-eTOM> (Accessed: 10.06.2025).
 14. Dovhyi S. O., Kopiika O. V. *IT-infrastruktura yak bazova skladova tsyfrovoyi transformatsii*. Monohrafiia. Kyiv: Yuston. 2023. 458 s.
 15. Bezpechna korporativna arkhitektura: TOGAF ta ArchiMate dlia staloho zrostannia. URL: <https://surl.lu/vnytek> (Accessed: 10.06.2025).
 16. Oliieshko D. Povnyi hid z 25 naikrashchykh prohrannykh instrumentiv dlia obsluhovuvannia kliientiv. URL: <https://surl.li/spjmjb> (Accessed: 10.06.2025).
 17. Bosu Michael, Abuaiadah Diab et al. Evaluation of IT Service Desk: A Case Study. URL: https://www.academia.edu/70472216/Evaluation_of_IT_Service_Desk_A_Case_Study (Accessed: 10.06.2025).
 18. Burian S. S., Pohorielovska I. D. Intehrovana systema sluzhby tekhnichnoi pidtrymky na osnovi Telegram i Jira Service Desk. *Ekonomika i biznes: novi realii, tekhnolohichni trendy, tsyfrovi innovatsii: zbirnyk materialiv Mizhnarodnoi*

- naukovo-praktychnoi konferentsii здобувачів вищої освіти та молодих вчених (Lviv–Irpin, 15 травня 2025 р.). Lviv: LNU im. Ivana Franka. 2025. S. 530–533.
19. Ukrainski media, stavlennia ta dovira u 2024 r. USAID-Internews survey. URL: https://drive.google.com/file/d/1kwsclr3Qm2QaqaIVv0I4saWRFY_NXWb/view?pli=1 (Accessed: 05.09.2025).
 20. Telegram FAQ. URL: <https://telegram.org/faq> (Accessed: 23.07.2025).
 21. Telegram Bot API Server. URL: <https://github.com/tlib/telegram-bot-api> (Accessed: 23.07.2025).
 22. Atlassian Corporation. Jira REST API Documentation. URL: <https://developer.atlassian.com/cloud/jira/platform/> (Accessed: 07.07.2025).
 23. OWASP Top 10 API Security Risks – 2023. URL: <https://owasp.org/API-Security/editions/2023/en/0x11-t10/> (Accessed: 05.09.2025).
 24. Voloshyn O. F. *Modeli ta metody pryiniattia rishen.* /O. F. Voloshyn, S. O. Mashchenko. 2-e vid., pererob. ta dopov. K.: Vidavnychopolihrafichii centr "Kyivskyi universytet", 2010. 336 s.
 25. Teorey T. J., Morgan T. Et al. Database Design: Know It All. (Morgan Kaufmann Know It All) : Morgan Kaufmann, 2008. 368 p.
 26. Fielding R. T., Taylor R. N. Principled design of the modern Web architecture. In Proceedings of the 2000 International Conference on Software Engineering (ICSE 2000), Limerick, Ireland, June 2000. P. 407–416.
 27. Zhurovskyi M. Z., Pankratova N. D. *Osnovy systemnoho analizu.* K.: Vidavnychopolihrafichii centr "Kyivskyi universytet", 2007. 544 s.
 28. Lytvyn V. V., Pasichnyk V. V., Yatsyshyn A. V. *Systemy shthynnoho intelektu.* Lviv: "Mahnoliia 2006", 2017. 676 s.

Pogorielovska I.D., Burian S.S. Directions for optimizing IT service management processes

Standards, methodologies and practices of technical support services are considered. The problems of the efficiency of traditional technical support channels in modern companies are analyzed, an innovative hybrid solution is proposed based on the integration of Telegram Bot with Jira Service Desk. Architectural solutions, algorithms of the system operation and features of the implementation of online interaction between users and support service specialists are highlighted. A three-stage security architecture is developed with division into Internet zones, a demilitarized zone and an internal corporate network. Algorithms of authorization, distribution of applications and processing of online sessions are proposed, taking into account the workload of specialists and a multi-criteria approach. A database is designed with a minimum set of information without confidential business data, which meets modern principles of information security and provides quick access to the necessary information. Caching mechanisms are used to reduce the load on internal systems and increase the overall productivity of the solution. The proposed online consultation mechanism transforms the traditional asynchronous request management system into a tool for real-time operational interaction, which is critically important in remote work conditions. The ability to instantly contact a specialist and solve a problem in a live dialogue mode significantly increases the efficiency of technical support and user satisfaction. The practical value of the developed system lies in the possibility of its implementation in organizations of various sizes and business profiles. The developed system is scalable, secure and provides all the necessary capabilities for the effective organization of technical support for users. This solution is especially relevant for Ukrainian companies in the context of modern realities of remote work and the need to ensure the continuity of business processes.

Keywords: IT service management, systems integration, software architecture, Telegram Bot API, Jira Service Desk.

Погорєловська Ірина Дем'янівна, кандидат економічних наук, доцент, доцент кафедри комп'ютерних та інформаційних технологій і систем Державного податкового університету, i.d.pogorielovska@dpu.edu.ua

Бур'ян Сергій Сергійович, здобувач вищої освіти другого (магістерського) рівня факультету фінансів та цифрових технологій Державного податкового університету, buriansereja@gmail.com