

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління

ВОЛКОВА Анастасія Сергіївна

**Модель управління командою підтримки виконання
замовлень ІТ-компанією / IT Company Order Fulfillment
Support Team Management Model**

спеціальність: 122 - Комп'ютерні науки
освітньо-професійна програма - Комп'ютерні науки

Кваліфікаційна робота

Виконала студентка групи
КНм-21
А.С. Волкова

Науковий керівник:
к.т.н., доцент Н.М. Васильків

Кваліфікаційну роботу
допущено до захисту:
«___» _____ 20___ р.
Завідувач кафедри
_____ М.П. Комар

ТЕРНОПІЛЬ - 2022

Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління
Освітній ступінь «магістр»
спеціальність: 122 – Комп'ютерні науки
освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри
М.П. Комар
«_____» _____ 20__р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТЦІ
Волковій Анастасії Сергіївні
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи

Модель управління командою підтримки виконання замовлень ІТ-компанією /
IT Company Order Fulfillment Support Team Management Model
керівник роботи к.т.н., доцент Н.М. Васильків
затверджені наказом по університету від 31 грудня 2021 року № 606.

2. Строк подання студентом закінченої кваліфікаційної роботи 16 листопада 2022 року.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити

- характеристика процесу надання послуг клієнтам командою підтримки ІТ-компанії;
- аналіз відомих моделей управління командами підтримки;
- аналіз схем та алгоритмів багаторівневої моделі та моделі «роїння»;
- постановка задачі дослідження;
- схема та алгоритм комбінованої моделі управління командою підтримки виконання замовлень;
- математичний опис та порівняння багаторівневої моделі, моделі «роїння» та комбінованої моделі;
- нечітка систему оцінювання впливу моделі команди підтримки на виконання змовлень;
- дослідження запропонованої комбінованої моделі на прикладі реальної компанії та аналіз отриманих результатів.

5. Перелік графічного матеріалу у роботі

- схеми багаторівневої моделі та моделі «роїння»;
- схеми алгоритмів багаторівневої моделі та моделі «роїння»
- схема комбінованої моделі управління командою підтримки;

РЕЗЮМЕ

Кваліфікаційна робота на тему «Модель управління командою підтримки виконання замовлень ІТ-компанією» на здобуття освітнього ступеня «Магістр» зі спеціальності 122 «Комп'ютерні науки» освітньої програми «Комп'ютерні науки» написана обсягом в 99 сторінок і містить 17 ілюстрацій, 1 таблицю, 4 додатки та 47 використаних джерел.

Метою даної кваліфікаційної роботи є розробка моделі управління командою підтримки виконання замовлень на основі дослідження та аналізу сучасних моделей управління командами ІТ-підтримки.

Методи досліджень: математичне моделювання, нечітка логіка, методологія створення програмних продуктів.

Результати дослідження: розроблено комбіновану модель управління командою підтримки виконання замовлень ІТ-компанією, в якій здійснюється попередня типізація запитів і, відповідно, пришвидшується виконання завдань; побудовано нечітку систему оцінювання впливу команди підтримки на виконання замовлень.

Результати розробленої комбінованої моделі можуть успішно застосовуватися для ефективного управління командами підтримки в ІТ-компаніях при обслуговуванні клієнтів та вирішенні вхідних запитів.

Ключові слова: КОМАНДА ІТ-ПІДТРИМКИ, БАГАТОРІВНЕВА МОДЕЛЬ, МОДЕЛЬ РОЇННЯ, КОМБІНОВАНА МОДЕЛЬ, НЕЧІТКА СИСТЕМА.

ABSTRACT

The qualification work on the topic " IT Company Order Fulfillment Support Team Management Model" for the Master's degree on speciality 122 "Computer Science" educational and professional program "Computer Science" is written on 99 pages and it contains 17 figures, 1 table, 4 annexes and 47 sources.

The purpose of this qualification work is to develop an order fulfillment support team management model based on research and analysis of modern IT support team management models.

Research methods: mathematical modeling, fuzzy logic, methodology for creating software products.

Research results: a combined model of order support team management by an IT company was developed, in which preliminary typification of requests is carried out and, accordingly, the execution of tasks is accelerated; a fuzzy evaluation system of the support team's impact on order fulfillment was built.

The results of the developed combined model can be successfully applied to the effective management of support teams in IT companies when serving customers and solving incoming requests.

Key words: IT SUPPORT TEAM, TIERED MODEL, SWARMING MODEL, COMBINED MODEL, FUZZY SYSTEM.

ЗМІСТ

Вступ	7
1 Управління командою підтримки ІТ-компанії	10
1.1 Надання послуг клієнтам командою підтримки ІТ-компанії	10
1.2 Аналіз відомих методів та рішень з управління командою підтримки клієнтів.....	15
1.3 Постановка задачі дослідження.....	29
Висновки до розділу 1	30
2 Моделі управління командою підтримки виконання замовлень ІТ-компанії.....	31
2.1 Традиційна модель управління командою ІТ-підтримки.....	31
2.2 Модель «роїння».....	42
2.3 Комбінована модель ІТ-підтримки	52
Висновки до розділу 2	57
3 Дослідження комбінованої моделі ІТ-підтримки.....	58
3.1 Дослідження на основі математичного моделювання.....	58
3.2 Нечітка система оцінювання впливу команди підтримки на виконання замовлень	62
3.3 Практичне застосування комбінованої моделі.....	71
Висновки до розділу 3	79
Висновки	80
Список використаних джерел	82
Додаток А База правил нечіткої системи.....	88
Додаток Б Текст коду	89
Додаток В Довідка про використання результатів дослідження	90
Додаток Г Копії опублікованих матеріалів дослідження	91

ВСТУП

Актуальність. Сьогодні, як ніколи, компанії всіх типів залежать від інформаційних технологій для досягнення своїх бізнес-результатів, незалежно від того, до якої сфери належить їх діяльність. Наприклад, якщо виробництво, хоча б частково, зупиняється через помилку чи збій у роботі програмного забезпечення, то очікується, що ІТ-спеціалісти швидко відреагують і вирішать цю проблему. Оскільки підприємства дуже залежать від програмних додатків та ІТ-послуг, прості стають дедалі неприйнятними. Підприємства все більше підвищують вимоги для ІТ-служб, коли справа доходить до доступності ІТ-сервісів, та й ІТ-служби повинні надавати надійні послуги з високою доступністю.

Крім того, стає очевидним попит на більш високу якість ІТ-послуг, що надаються. Будь-які дефекти чи збої в ІТ-продуктах впливають на задоволеність клієнта.

В основі ІТ-послуг лежить фактична послуга, яка має надати клієнтові цінність і допомогти йому досягти конкретної мети, а для цього і потрібна служба підтримки.

Таким чином, щоб ІТ-компанія могла добре обслуговувати своїх клієнтів, обидві сторони мають чітко узгодити цілі та точний характер послуг, що надаються.

Коли бізнес сильно залежить від ІТ-інструментів і послуг, вплив підтримки легко відчутти, незалежно від того, чи є вона внутрішньою чи зовнішньою. Зовні хороша ІТ-підтримка, як правило, у формі обслуговування клієнтів, призведе до збільшення прибутку та утримання клієнтів. На внутрішньому рівні хороша ІТ-підтримка сприяє підвищенню задоволеності та продуктивності працівників.

Хороша ІТ-підтримка призводить до таких переваг для бізнесу:

- підвищення рівня задоволеності клієнтів або співробітників;
- підвищення продуктивності;
- виявлення проблем, що в результаті призводять до підвищення якості ІТ-продуктів/послуг;

- визначення можливостей покращення бізнес-процесів.

Саме для забезпечення ефективної та якісної ІТ-підтримки існують різні моделі управління командами підтримки.

Вплив поганого обслуговування клієнтів у галузі ІТ-послуг є далекосяжним. Це впливає не лише на бізнес, але й на клієнтів компанії і, у свою чергу, на їхніх клієнтів. Клієнти, швидше за все, звернуться до компанії за повторною послугою, оскільки вони задоволені попередньою співпрацею. В ІТ-індустрії, де майже всі послуги надаються на основі контракту, оновлення існуючих контрактів стає найголовнішим фактором успішності ІТ-компанії. Погане обслуговування клієнтів гарантує, що вони не продовжать свої контракти та звернуться до конкурентів. Тому розробка та дослідження моделі управління командою підтримки ІТ-компанії є актуальною задачею.

Мета і завдання роботи. Метою роботи є розробка моделі управління командою підтримки виконання замовлень на основі дослідження та аналізу сучасних моделей управління командами ІТ-підтримки.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- проаналізувати відомі моделі управління командами підтримки в ІТ-компаніях: традиційну багаторівневу модель та модель «роїння»;
- виділити переваги та недоліки обох моделей, а також труднощі пов'язані з їх імплементацією та використанням;
- розробити комбіновану модель управління командою підтримки виконання замовлень, яка ґрунтуватиметься на основі переваг обидвох моделей;
- здійснити математичний опис та на його основі порівняння багаторівневої моделі, моделі «роїння» та комбінованої;
- розробити нечітку систему оцінювання впливу моделі команди підтримки на виконання замовлень;
- дослідити запропоновану комбіновану модель на прикладі реальної компанії та проаналізувати отримані результати.

Об'єкт дослідження – виконання замовлень ІТ-компанією.

Предмет дослідження – моделі управління командою підтримки виконання замовлень ІТ-компанії.

Методи досліджень: математичне моделювання, нечітка логіка, методологія створення програмних продуктів.

Наукова новизна одержаних результатів:

Розроблено комбіновану модель управління командою підтримки виконання замовлень ІТ-компанією, в якій, на відміну від існуючих, здійснюється попередня типізація запитів і, відповідно, пришвидшується виконання завдань.

Практичне значення отриманих результатів.

Розроблена комбінована модель може бути використана для ефективного управління командами підтримки в ІТ-компаніях при обслуговуванні клієнтів та вирішенні вхідних запитів.

Публікації та апробація. Результати досліджень опубліковані у двох збірниках матеріалів міжнародної наукової інтернет-конференції «Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення»: випуск 66 (м. Тернопіль, 6-7 квітня 2022 року) та випуск 72 (м. Тернопіль, Україна – м. Переворськ, Польща, 15-16 листопада 2022 року).

1 УПРАВЛІННЯ КОМАНДОЮ ПІДТРИМКИ ІТ-КОМПАНІЇ

1.1 Надання послуг клієнтам командою підтримки ІТ-компанії

ІТ-індустрія є сектором, який швидко розвивається, і, ймовірно, він продовжить розвиватися. Дохід, отриманий від продуктів/послуг у цій сфері, є наймовірно високим і з роками зростає через попит на них з боку споживачів у всьому світі.

Зараз ІТ-сфера – це всеохоплююча галузь, будь-яка робота, яка, в основному, пов'язана з роботою комп'ютерів або розробкою для них, належить до ІТ-індустрії/сектору.

ІТ-продукти та ІТ-послуги – це два терміни, які в сучасному світі використовуються як синоніми. Вони стосуються будь-якого типу програмного або апаратного забезпечення, яке полегшує створення цифрового вмісту та обмін ним у комп'ютерних мережах, або будь-якої іншої технології, яка допомагає полегшити життєдіяльність людей.

Визначення відмінностей між ІТ-продуктами та ІТ-послугами може змінити розуміння цільової аудиторії. Хоча ці дві категорії мають певну схожість, вони також відрізняються в кількох ключових аспектах. Ці відмінності впливають на те, хто може бути зацікавлений у придбанні певних продуктів, а не в самих послугах.

Великі ІТ-компанії створюють власні продукти та надають послуги клієнтам цих продуктів. Прикладами таких компаній є Microsoft, Oracle, Google тощо. Однак ІТ-компанія, що надає послуги, зазвичай використовує ІТ-продукти інших компаній та пропонує послуги клієнтам базуючись на можливостях та функціях зовнішнього програмного забезпечення, яке вона використовує [1].

Все частіше бізнеси хочуть мати персоналізовані функції або унікальні додатки, створені спеціально під особливості компанії, тому розробка програмного забезпечення на замовлення є новою тенденцією у світі технологій і вона швидко поширюється серед основних компаній у всьому світі. У світі, де підприємствам стає все складніше виділитися та бути оригінальними, такі послуги користуються наймовірним попитом.

Компанії з розробки програмного забезпечення на замовлення – це бізнес-організації, які створюють та налаштовують програмне забезпечення відповідно до конкретних потреб своїх клієнтів. В Україні більшість ІТ-компаній виконують замовлення за принципом аутсорсингу [2].

Аутсорсинг – це угода, за якою одна компанія наймає іншу компанію для відповідальності за заплановану або існуючу діяльність, яка виконується або може здійснюватися всередині компанії, і іноді передбачає передачу працівників та активів від однієї фірми до іншої.

Існує принаймні чотири види аутсорсингу програмного забезпечення:

1) оншорна розробка програмного забезпечення – компанії замовника та виконавця розташовані в тій самій країні (часто також у тому самому місті). Один часовий пояс, одна мова, зазвичай один менталітет – це все в рази полегшує співпрацю;

2) офшорна розробка програмного забезпечення – компанії, що виконують розробку ПЗ для замовників за кордоном;

3) близький аутсорсинг програмного забезпечення – це коли компанія передає роботу з розробки програмного забезпечення в сусідню країну, яка розташована у схожих часових поясах, що дуже спрощує комунікацію;

4) аутсорсинг гібридного типу – проекти з оншорним управлінням та командами розробників в офшорних та ближніх зонах.

Кожен підхід, безсумнівно, має свої плюси і мінуси. Зокрема, оншорний підхід пропонує високу якість та частоту особистих зустрічей і добре підходить для проектів на 4-12 місяців із великим бюджетом. А офшорна розробка є вигідною для компаній з чітко визначеними проектами, які шукають економічно ефективніше рішення [3].

Послуги з розробки програмного забезпечення на замовлення включають процеси, у яких компанія налаштовує різні потреби програмного забезпечення для своїх клієнтів, що означає, що вони можуть надавати продукти та послуги, спеціально розроблені для них, а це у свою чергу призводить до нижчих витрат і вищої якості порівняно з конкурентами, які не можуть запропонувати цю послугу

або сервіс.

Спеціальне програмне забезпечення створене для задоволення унікальних потреб бізнесу та використовує засоби програмування, фреймворки, бібліотеки та інші інструменти залежно від вимог проекту для створення індивідуальних рішень для своїх клієнтів, наприклад:

- бази даних, створені відповідно до потреб та запитів компанії;
- спеціальні кампанії електронною поштою для втілення унікальних маркетингових цілей;
- індивідуальні рішення IoT, розроблені спеціально для певної галузі, як-от розумні будинки чи товари для охорони здоров'я.

Завдяки тому, що компанії з розробки програмного забезпечення на замовлення пропонують унікальні та оригінальні рішення, бренд-замовник завжди отримуватиме інноваційний продукт, який значно відрізнятиметься від продуктів його конкурентів.

В основі надання таких послуг лежить принцип виконання конкретного завдання краще, швидше та ефективніше, ніж це можливо самостійними силами замовника.

Також, такі компанії мають великий досвід надання послуг комплексної розробки програмних продуктів для замовників, у яких недостатньо власних ресурсів для реалізації проекту. Вони можуть з легкістю інтегрувати своїх фахівців у вже функціонуючі проектні команди клієнтів або повністю взяти на себе всі технологічні функції залежно від завдання, поставленого замовником.

Компанії можуть пропонувати повний цикл послуг із розробки програмного забезпечення, який включає всі етапи циклу розробки або ж часткові зміни, коли потрібно додати новий функціонал у старий проект.

У наш час із переходом до цифрового бізнесу технічна компетенція команди підтримки є набагато більш поширеною, тому компанії, які займаються розробкою програмного забезпечення, зазвичай пропонують не звичайну клієнтську підтримку, а постійну технічну підтримку 24/7, тому вирішити будь-які проблеми швидко та ефективно буде неважко [2].

Технічна підтримка відіграє одну з ключових ролей у бізнесі, а також через спілкування з клієнтами приносить нові ідеї для розробки продуктів, їх покращення та управління даними.

Обслуговування клієнтів полягає у ставленні компанії до своїх клієнтів з повагою, турботою та встановленні з ними емоційного зв'язку. Це те, чим може (і повинен) займатися кожен у команді, а не лише представник служби підтримки клієнтів чи менеджер із успіху клієнтів.

Відмінність технічної підтримки полягає у тому, що вона зосереджена на вирішенні технічної проблеми найшвидшим та найефективнішим способом в той час, коли ж звичайне обслуговування клієнтів натомість зосереджується на досвіді клієнта.

Працівники технічної підтримки не обов'язково є програмістами, а (всупереч поглядам багатьох людей, які не займаються ІТ) інженери програмного забезпечення не обов'язково добре володіють технічною підтримкою.

Представники технічної підтримки уважно відслідковують ознаки проблеми, намагаються її відтворити та швидко знаходять та пропонують її вирішення. У більшості ситуацій у роботі технічної підтримки основним показником успіху є те, що клієнту не потрібно знову звертатися до служби підтримки.

Обслуговування клієнтів гарантує, що агенти вирішують проблеми клієнтів, одночасно підтримуючи їхні емоційні потреби. Клієнтський досвід охоплює весь шлях клієнта до бренду, від знайомства з послугою та порівняння її з іншими конкурентними варіантами на ринку до завершення співпраці, що може включати, а може і не включати звернення до служби підтримки клієнтів або отримання допомоги.

Обслуговування клієнтів – це більше, ніж просто надання послуг, які споживачі очікують від компанії, або надання належної технічної підтримки. Тут також йдеться про задоволення їхніх емоційних потреб і зміцнення стосунків між клієнтом та бізнесом. Для цього менеджери команди підтримки повинні ставитися до клієнтів так, як вони хочуть, щоб ставилися до них в аналогічних ситуаціях. Компанії зобов'язані прислухатися до потреб кожного споживача і знайти

найкраще рішення для вирішення його запиту.

Робота команди підтримки обслуговування клієнтів – це пошук того, що працює для клієнта – не лише рішення проблеми чи запиту, яке він шукає, але й тип взаємодії та ставлення, які найкраще підходять конкретному клієнту. Споживач повинен відчувати, що він працює з партнером, який розуміє проблему з його точки зору.

Вкрай важливо мати професійні команди обслуговування клієнтів, оскільки будь-яка взаємодія зі службою підтримки може раптово стати проблемою для ІТ-проекту. Оскільки незадоволені клієнти, швидше за все, поділяться своїм досвідом з іншими, один негативний випадок обслуговування клієнтів може зашкодити репутації всієї ІТ-компанії, навіть якщо агенти підтримки здебільшого надають швидку, професійну і якісну допомогу. Якщо акцентувати увагу на якісному обслуговуванню клієнтів та робити це пріоритетом бізнесу, то створюється довіра клієнтів, зменшується відтік незадоволених користувачів і збільшується прибуток проекту [1].

Коли ІТ-компанії роблять інвестиції у турботу про клієнтів, то таким чином вони створюють тривалу лояльність до бренду зі сторони споживача та підвищують задоволеність клієнтів від користування їхніми послугами та продуктами. Якщо клієнтам комфортно користуватись послугами та вони отримують задоволення від обслуговування, то вони, швидше за все, довірятимуть цій компанії. Якісне обслуговування клієнтів є джерелом життя будь-якого бізнесу, бо створює позитивний досвід для користувача, що призводить до повторних ділових зв'язків, угод і рекомендацій.

А для того, щоб якісно обслуговувати клієнтів, потрібно якісно керувати ІТ-послугами. Управління ІТ-послугами – це впровадження та управління якісними ІТ-послугами, які відповідають потребам бізнесу та здійснюються постачальниками ІТ-послуг за допомогою відповідного поєднання людей, процесів та інформаційних технологій.

Оскільки з часом та ростом ІТ-компанії продажі зростають, канали взаємодії з клієнтами розмножуються, продукти додаються до асортименту та розширюється

клієнтська база, можуть виникати потенційні перешкоди. Отже, в такому випадку, система керування ІТ-послугами може допомогти усунути вразливі місця в міру зростання та розвитку ІТ-компанії.

Бібліотека інфраструктури інформаційних технологій (ITIL) – це система найкращих практик і рекомендацій щодо управління ІТ-операціями та послугами організації. ITIL дає вказівки щодо надання якісних ІТ-послуг і процесів, функцій та інших можливостей, необхідних для їх підтримки. Фреймворк ITIL базується на життєвому циклі сервісу та складається з п'яти етапів життєвого циклу (стратегія сервісу, проектування сервісу, перехід до сервісу, функціонування сервісу та постійне вдосконалення сервісу), кожен з яких має власну супровідну публікацію. Його замовило Центральне агентство комп'ютерів і телекомунікацій уряду Великобританії (ССТА) у середині 1980-х років. Процеси ITSM, побудовані на базі ITIL, прокладають шлях до кращих ІТ-послуг і вдосконалення бізнесу.

Незалежно від розміру бізнесу, кожна організація певним чином залучена до управління ІТ-послугами. Управління ІТ-послугами забезпечує спрощене керування інцидентами, запитами на обслуговування, проблемами, змінами та ІТ-активами на додаток до інших аспектів ІТ-послуг. ІТ-команди організації можуть використовувати різні робочі процеси та передові методи управління ІТ-послугами, як зазначено в ITIL [4].

1.2 Аналіз відомих методів та рішень з управління командою підтримки клієнтів

Служба підтримки компанії – це об'єднання людей, обладнання, розгорнутих програмних агентів та програмних засобів, які підтримують правильне функціонування та представлення послуг компанії для клієнтів. Ця служба повинна бути інтегрована із справною апаратною та програмною системами моніторингу усіх послуг, які пропонує компанія.

У [5] автори описують базову архітектуру служби підтримки на одній з американських компаній.

Запропонована архітектура команди підтримки намагається вирішити 90% запитів клієнтів протягом перших 20 хвилин контакту і в середньому цій моделі потрібно менше ніж 15 хвилин для вирішення усіх внутрішніх помилок у разі їх виявлення. Ця служба підтримки складається з чотирьох основних рівнів і трьох окремих груп агентів:

- рівень 0: самодопомога клієнта. Клієнту надається допомога у формі доступу до репозиторію знань, поширених запитань і діагностичного програмного забезпечення. Неможливість вирішення питань на цьому рівні призводить до залучення служби підтримки;

- рівень I: клієнтська підтримка та довідкова служба. Клієнт може попросити допомогу за номером телефону служби підтримки, електронною поштою або використовуючи веб-форми. Це підрозділ, який розроблятиме звіт про несправності і доводитиме більшість запитів до повного вирішення. Рівень I є точкою входу клієнта у систему підтримки. На цьому рівні клієнт звертається у службу підтримки щодо проблеми чи іншого питання, яке могло виникнути у нього упродовж користування сервісом. Агент служби підтримки збирає від клієнта всю необхідну інформацію, як-от ім'я, адресу, номер соціального страхування тощо, перевіряє особу клієнта та намагається швидко вирішити проблему за допомогою попередньо визначеного сценарію та доступу до репозиторію знань (бази знань). Якщо це не вирішує проблему, вона класифікується та передається на відповідний рівень (рівень II);

- рівень II: проактивний моніторинг послуг. Цей блок відстежуватиме мережеву діяльність і продуктивність служб за допомогою інструменту тестування та ряду вбудованих програмних засобів. Він також може генерувати звіти про несправності перед тим, як клієнти сконтактують з командою підтримки, щоб повідомити про проблеми. На рівні II обізнаний агент проходить за більш детальним сценарієм для збору інформації від клієнта та використовує процес моніторингу на основі програмного забезпечення, щоб отримати більш детальну інформацію про ситуацію. Персонал рівня II має більше доступу та привілеїв, що дозволяє їм вносити зміни до системи та вирішувати проблеми. Якщо персонал

рівня II зміг вирішити проблему, він закриває звіт про несправність і оновлює репозиторій знань інформацією про вирішення несправності. Якщо проблему неможливо вирішити на цьому рівні, звіт про несправність буде переведено на рівень III;

- рівень III: активний моніторинг. Цей підрозділ відстежуватиме всі сповіщення про помилки, отримані агентами I та II рівнів, і на основі методів та практик, розроблених у компанії, надсилатиме групу спеціалістів для фізичної оцінки проблем над якими працюють агенти попередніх рівнів. Цей рівень допоможе рівню I і рівню II, коли ті не можуть вирішити проблему. На рівні III проводиться поглиблений аналіз і робота з кодом. Якщо це вирішить проблему, будь-які помилки коду буде надіслано команді розробників, яка займатиметься виправленням та керуванням конфігурацією продукту, та тимчасове швидке виправлення буде створено для майбутніх проблем і додано до репозиторію знань.

Для того, щоб полегшити роботу всіх трьох підрозділів (I, II та III рівнів), необхідні набори спеціального програмного забезпечення на комп'ютерах агентів, добре структурована система репозиторію знань та правильний опис попередніх сценаріїв. Ці засоби мають бути добре інтегрованими з даними моніторингу стану апаратного забезпечення та операційних систем агентів. З цієї причини всі три підрозділи використовуватимуть програмне забезпечення для моніторингу послуг, яке має бути гнучким, легко налаштовуватись, розширюватись та адаптовуватися до нової інформації, яка надходить з моніторингу всієї системи.

Кожен підрозділ матиме свого адміністратора з певними обов'язками. Кожна зміна матиме такого адміністратора і команда адміністраторів регулярно зустрічатиметься з тою періодичністю, яка буде продиктована подіями, які відбуваються у системі, щоб перевіряти діяльність агентів та змінювати або створювати нові методи та практики взаємодії між рівнями підтримки. Для певних дій у системі потрібні права адміністратора. Весь персонал служби підтримки повинен дотримуватися політики безпеки компанії, включаючи двосторонню автентифікацію та авторизацію для контролю доступу до функцій системи.

Репозиторій знань – це єдине інтегроване джерело всієї інформації про роботу підприємства. Він оновлюватиметься всіма трьома підрозділами в рамках служби підтримки і буде доступний для всіх трьох підрозділів у межах цієї служби. Агенти оновлюють базу даних за розкладом або на вимогу. Репозиторій складається з бази даних та інструментів для імпорту та категоризації даних, а також інструментів для пошуку. Доступ до репозиторію контролюється на всіх рівнях [5].

У [6] автори стверджують, що компанія може бути більш прибутковою, якщо вона використовує багаторівневу модель управління службою підтримки. Вони вважають, що багаторівнева структура допомагає компанії краще контролювати витрати на відшкодування коштів за послуги шляхом перевірки менш суворих претензій, щоб такі клієнти не зверталися до спеціалістів на вищих рівнях і перевірки усіх нелегітимних претензій (тих, що виникли не за виною компанії).

Щоразу, коли клієнт купує продукт або послугу, існує ймовірність незадоволення. Незалежно від причини невдоволення, клієнт може звернутися до організації обслуговування клієнтів, щоб вимагати відшкодування. Ця служба може бути у формі онлайн-чату з агентом або кол-центру.

Багато з цих служб підтримки, наприклад офшорні кол-центри, характеризуються багаторівневою організаційною структурою. Наприклад, кол-центри, які обслуговують компанію Dell, мають набір агентів «Рівня І», які приймають початкове звернення від клієнта. Ці агенти навчені надавати стандартизовані варіанти вирішення проблеми та мають обмежені повноваження щодо надання компенсації за продукт чи послугу. Якщо проблему клієнта не вирішено задовільним чином, він може передати свою претензію більш високопоставленому агенту, який знаходиться на рівень вище, наприклад керівнику структурного підрозділу служби підтримки, який має право надавати більшу компенсацію.

Однак для багатьох клієнтів спілкування з агентами підтримки забирає багато часу та викликає розчарування. У США споживач витрачає в середньому 13 годин на рік у чергах для дзвінків та зазвичай звертається кілька разів з приводу однієї

проблеми, перш ніж його вимога в компенсації буде задоволена, або й взагалі відмовляється від ідеї отримання відшкодування. Час і зусилля, витрачені на цей процес, означають, що незадоволеним клієнтам потрібно зазнавати зайвого клопоту та витраченого часу, щоб вимагати компенсацію коштів.

На основі отриманих результатів [6], автори припускають, що для фірми можуть бути вигідні такі клопоти клієнта. Автори виявили, що запровадивши багаторівневу модель підтримки, компанія може уникнути надто великих відшкодувань, зокрема, організація може не мати переконливого стимулу повністю усунути клопоти клієнтів, навіть якщо це було б можливо з операційної точки зору.

Автори вивчили мікроекономічні стимули незадоволеного клієнта, який домагається відшкодування певних коштів за послугу від служби підтримки. Зокрема, вони розробили модель процесу розгляду скарг, у якій претензії клієнтів розглядаються та оцінюються командою підтримки.

У їхній моделі фірма вказує лише ліміт, який має право виплачувати будь-який агент підтримки рівня I. Результат демонструє, що ці ліміти виплат можуть бути визначені таким чином, щоб змусити незадоволеного клієнта пройти послідовний процес подання претензій, подібний до «стрибків через кільця», щоб отримати компенсацію. Вибір компанією цих обмежень передбачає багаторівневу структуру, яка вимагає від будь-якого незадоволеного клієнта спочатку озвучити скаргу представнику підтримки першого рівня, який не має повноважень пропонувати вищий розмір відшкодування. У випадку, якщо клієнт вважає цю пропозицію заниженою, він зазнає клопоту та витрачання зайвого часу, пов'язаного з перенесенням своєї претензії до представника підтримки вищого рівня, який має право надавати компенсацію на вищу суму.

Автори доводять, що багаторівнева структура зменшує витрати компанії на відшкодування, відсіюючи претензії, які є менш суворими, менш вмотивованими та нелегітимними. Наприклад, деякі клієнти з менш серйозними скаргами вважають додатковим клопотом та тратою часу спілкування з представниками вищих рівнів підтримки. Тобто, щоб вирішувати проблеми з клієнтами щодо відшкодування коштів, багаторівнева підтримка відсіює менш суворі претензії

таким чином, що деякі клієнти зупиняються на першому рівні й отримують нижчі суми відшкодування.

Багаторівнева підтримка також може допомогти перевірити незаконні претензії, щоб клієнти без належного обґрунтування мали меншу ймовірність отримати вищу компенсацію.

Наприклад, користувач може неправомірно стверджувати, що його продукт вийшов з ладу через дефекти виробництва, хоча насправді це було спричинено його власним неправильним використанням.

Під час першого звернення до служби підтримки, клієнту пропонують невелику суму компенсації без повної перевірки причини збою. Відшкодування у більшій сумі можливе тільки у тому випадку, якщо клієнт зможе обґрунтувати свою вимогу.

Таким чином, клієнти з нелегітимними претензіями мають меншу ймовірність перенести претензію на вищий рівень і, отже, отримують нижчі виплати.

Іншими словами, використовуючи різницю у витратах між нелегітимними та законними претензіями, багаторівнева підтримка перевіряє претензії навіть без початкового спостереження за справжнім станом проблеми.

Допоміжною перевагою багаторівневої структури є те, що вона може додатково контролювати витрати на персонал, якщо від працівника другого рівня вимагається затвердити виплати відшкодування вищого рівня.

Використовуючи менш кваліфікованих (і дешевших) працівників на першому рівні, служба підтримки може відсіювати деякі претензії від співробітників вищого рівня, чий час є більш цінним.

Цей факт може допомогти пояснити тенденцію компаній делегувати більше повноважень офшорним кол-центрам або автоматизованим онлайн-системам підтримки клієнтів [6].

Автори [7] пропонують таку структурну організацію компанії, яка включає підрозділ підтримки клієнтів та взаємодію з ним, зображену на рисунку 1.1.

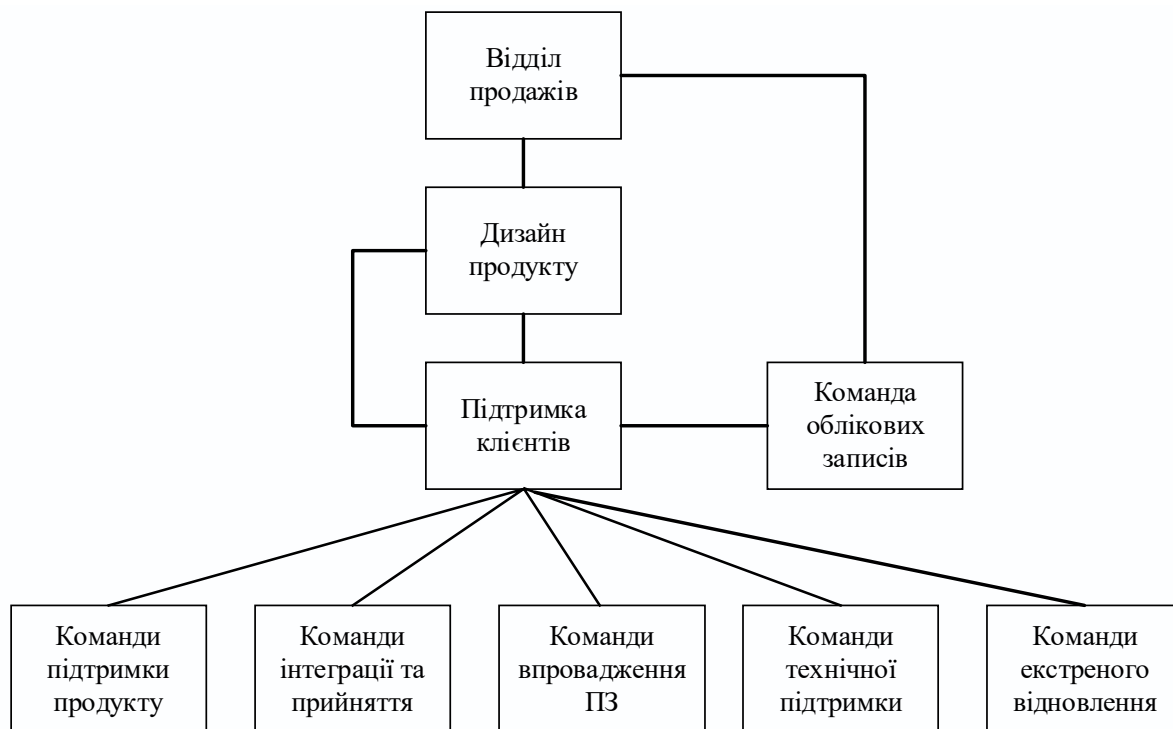


Рисунок 1.1 – Організація служби підтримки клієнтів

Команди підтримки клієнтів ІТ-компаній роблять свій внесок в організацію розробки (дизайну) продукту, надаючи інформацію для цих процесів. Маючи добре організовану, висококваліфіковану робочу силу, компанія є дуже універсальною та може надавати різноманітні послуги та рішення щодо продуктів та технологій, допомагаючи у дослідженнях, розробках, проектуванні, виправленні помилок та технічній підтримці.

Завдяки своїй критичній ролі у забезпеченні безперервності бізнесу, взаємодії між клієнтами та компанією, команди підтримки безпосередньо підпорядковуються керівнику всієї компанії. Компанія розміщує свої організаційні ресурси відповідно до потреб клієнтів у службі підтримки, щоб надавати кращі послуги та рішення, швидші процеси, мати міцніші стосунки з клієнтами, а, отже, й інноваційні продукти, які в кінцевому підсумку підвищують задоволеність та лояльність клієнтів.

На основі схеми, зображеної на рисунку 1.1, доцільно описати ролі команд служби підтримки клієнтів і параметри знань та обов'язків кожної з них. Також слід

вказати, що команди технічної підтримки є основними гравцями, які беруть участь у всіх рішеннях. Отже, характеристики команд є наступними:

- команди підтримки продукту – це інженери з більш глибокими знаннями та досвідом щодо конкретного продукту, які залучаються, якщо проблема надто складна або перевищує рівень інженера команди технічної підтримки. Вони також пропонують виправлення складних програмних проблем. Вони підтримують команди технічної підтримки та команди дизайнерів, надаючи експертні рішення для них і відгуки про необхідні вдосконалення функцій продукту для команд дизайнерів та розробників;

- команди інтеграції та прийняття – працюють безпосередньо з клієнтами на сайтах або ресурсах клієнтів під час встановлення, тестування та інтеграції нових систем;

- команди впровадження програмного забезпечення – допомагають клієнтам підтримувати програмне забезпечення в оновленому стані, виконуючи регулярні оновлення програмного забезпечення та виправлення помилок між оновленнями (патчі). Вони зв'язуються з командою технічної підтримки, якщо спостерігають будь-які проблеми до, під час або після оновлення програмного забезпечення;

- команди технічної підтримки – працюють 24/7 як перша та єдина точка контакту для вирішення проблем із клієнтським обладнанням або програмним забезпеченням, що спричиняє погіршення якості обслуговування або незначні збої на боці клієнта. Технічна підтримка зв'язується з командою підтримки продукту, якщо проблема виходить за межі рівня їх технічної кваліфікації та може спричинити серйозніші проблеми, або є надто складною, щоб її міг вирішити інженер технічної підтримки. Вони безпосередньо консультують команди інтеграції та прийняття, команди впровадження програмного забезпечення та клієнтів, маючи глибокі знання про широкий спектр продуктів і досвід, а також технічні навички та навички управління клієнтами, які можуть значно підвищити імідж компанії та репутацію служби підтримки клієнтів;

- команди екстреного відновлення – вирішують серйозні проблеми, через які є ризик спричинити ще більш серйозні проблеми у майбутньому, та працюють у

дуже жорстких часових рамках. На відміну від інших команд, вони можуть отримати допомогу від будь-якої частини компанії, щоб вчасно вирішити проблему. Після чого, команди технічної підтримки та підтримки продукту працюють разом, щоб надати клієнту аналіз першопричин та рішення, якщо проблема повториться.

До відома, ця архітектура також використовує репозиторій знань. Учасники команд можуть додавати короткі поради щодо технологій, як-от процедури, що часто використовуються, та додавати записані тренінги для навчання агентів різних команд у майбутньому [7].

Дослідження, про які йдеться у статті [8], ґрунтується на прикладі, проведеному у відділі технічного обслуговування користувачів системи з використанням підходу Six Sigma Define-Measure-Analyse-Improve-Control (Визначення-Вимірювання-Аналіз-Поліпшення-Контроль (DMAIC)) та його застосування для скорочення часу вирішення скарг. Стаття висвітлює практичне дослідження того, як методологія DMAIC допомогла скоротити час розгляду скарги з 12,5 до 8,5 годин. Це дослідження також призвело до скорочення часу виконання всіх основних процесів в компанії.

Структурований підхід до вирішення проблем DMAIC ставить такі питання процесу, як-от:

- визначення (Define) – у чому полягає проблема? Вона існує? Метою цього кроку є чітке озвучення проблеми, мети, потенційних ресурсів та масштабу проблеми;

- вимірювання (Measure) – як вимірюється процес? Як він проходить? Метою цього кроку є вимірювання специфікації проблеми. Це етап збору даних, метою якого є встановлення базових показників продуктивності процесу;

- аналіз (Analyse) – які найважливіші причини дефектів? Метою цього кроку є виявлення, перевірка та вибір першопричини проблеми для її усунення. Велика кількість потенційних першопричин проблеми визначається за допомогою аналізу першопричин та відбираються найвагоміші з них;

- поліпшення (Improve) – як можна усунути причини проблем? Метою цього кроку є визначення, тестування та реалізація вирішення проблеми;

- контроль (Control) – як можна підтримувати та супроводжувати поліпшення? Метою цього кроку є впровадження змін і забезпечення стабільності.

Компанія, описана у [8], є великою компанією з виробництва електрообладнання з штатом у 12 000 співробітників, розташована в Індії. Вони беруть участь у виробництві та введенні в експлуатацію електростанцій і пов'язаних з ними комунікацій, маючи клієнтську базу в різних країнах світу.

Компанія має систему планування ресурсів підприємства, запроваджену в усіх підрозділах компанії для всіх її видів діяльності. Повсякденна діяльність виконується за допомогою цієї системи. Усе, починаючи від замовлення матеріалів, закінчуючи виплатами зарплати працівникам, здійснюється за допомогою цієї системи планування ресурсів. Таким чином, існує постійна взаємодія із системою в різних відділах компанії.

Через великий розмір підприємства та характер його діяльності використання комп'ютера та мережі є дуже високим. Є відділ ІТ-підтримки, який займається проблемами комп'ютерів, мережі та програмного забезпечення в компанії. Щоразу, коли працівники стикалися з будь-якою проблемою, пов'язаною з апаратним чи програмним забезпеченням, мережею, вони негайно повідомляли про це ІТ-відділ електронною поштою, телефоном або вносили скаргу в онлайн-реєстр, який був у кожного структурного підрозділу. Після отримання скарги системний відділ доручав роботу одному зі своїх інженерів, і проблема вирішувалась. Оскільки від працівників компанії часто надходили скарги щодо затримки вирішення проблем, а компанія саме впроваджувала підхід DMAIC у весь свій виробничий процес, керівництво вирішило владнати цю проблему за допомогою методології DMAIC. Для вирішення цієї проблеми було сформовано групу з п'яти співробітників, в основному інженери з системного обслуговування та впровадження програмного забезпечення. В результаті визначено процес послідовності вирішення проблем, яким повинен був керуватися відділ підтримки користувачів зображений на рисунку 1.2 [8].



Рисунок 1.2 – Процес вирішення проблем

При використанні цієї моделі, команда підтримки повинна була провести роботу над кроками «Визначення» та «Вимірювання» проблеми, а інженерам ІТ-відділу потрібно було виконати етапи «Аналізу» та «Поліпшення». При цьому виконання етапу «Контроль» покладається на обидва відділи.

Автори [9] провели дослідження взаємодій, які виникають у сучасній практиці підтримки програмних продуктів. Вони представили результати багатовимірного аналізу діяльності з підтримки продуктів у провідній компанії, що займається розробкою програмного забезпечення. Автори провели кількісний аналіз наявних запитів на команду підтримки, опитування спеціалістів із підтримки продуктів, щоб зрозуміти поточну практику підтримки продуктів. Зокрема, дослідили корисність різних мультимедійних форматів, які можливі у сучасних веб-системах підтримки. Результати показали, що спеціалісти служби підтримки все ще стикаються з проблемами під час дистанційного вирішення проблем із програмним забезпеченням.

Компанія ABC, Inc. є провідною компанією по створенню програмного забезпечення, що спеціалізується на двовимірному та тривимірному проектуванні, інженерному та розважальному програмному забезпеченні для виробництва, проектування та будівництва, а також для медіа та розваг. Зараз на ринку представлено понад 80 різних продуктів ABC, які використовують понад 10 мільйонів кінцевих користувачів у 185 країнах.

Усі запити на підтримку реєструються у веб-системі відстеження (навіть ті, про які спочатку повідомлялося по телефону). Веб-система дозволяє клієнтам оновлювати інформацію та вкладати файли під час виконання запиту. Клієнти також можуть відстежувати хід виконання всіх своїх запитів через цю систему. Під

час надсилання запиту обов'язковими полями були назва продукту, операційна система (ОС), на якій цей продукт використовують, короткий коментар з одного речення та повний опис проблеми. Спеціалісти з підтримки продукту отримували доступ до запитів на підтримку через іншу веб-систему компанії, де зберігалася вся історія та вкладення, пов'язані з окремими запитами. Фахівці також можуть надсилати інформацію клієнтам, додавати оновлення та виконувати запити через цей інтерфейс.

Коли клієнт надсилає запит, його обробляють спеціалісти рівня I, які мають базові знання про продукти ABC, але не спеціалізуються на жодній конкретній сфері підтримки продукту. На рисунку 1.3 зображена модель служби підтримки ABC [9]. Вони можуть шукати відомі проблеми в консолідованій базі даних (репозиторій знань) і надавати допомогу, або вони можуть передати проблему спеціалісту з підтримки конкретного продукту, який знаходиться на рівень вище.

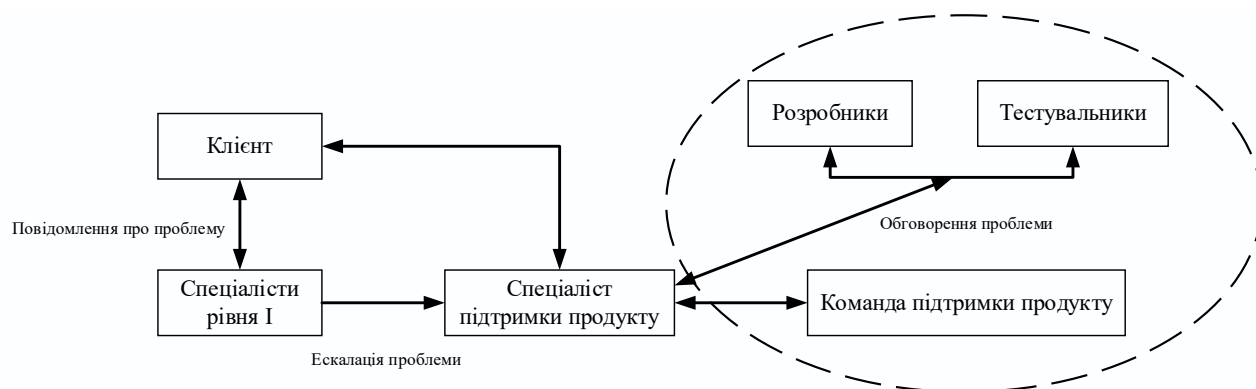


Рисунок 1.3 – Процес отримання запиту та його обробки

З цього моменту фахівець із служби підтримки продуктів і клієнт спілкуються один з одним напряму. Фахівці служби підтримки мають великий досвід роботи на системному та прикладному рівнях. За необхідності спеціалісти служби підтримки співпрацюють з іншими командами: розробниками чи спеціалістами з контролю якості (тестувальниками) для діагностики та вирішення складних проблем [9].

У статті [10] автори розглядають впровадження системи міркування на основі випадків (CBR (Case-based reasoning)) у компанії TISSAT, яка надає послуги технічної підтримки клієнтам різних компаній та організацій. Система CBR намагається вирішити проблему за допомогою повторного використання рішення старого подібного випадку. Це рішення попередньо зберігається в пам'яті випадків (базі випадків) і його можна або отримати та застосувати безпосередньо до поточної проблеми, або переглянути та адаптувати до нової проблеми.

В компанії є ряд технічних спеціалістів, чия роль полягає в наданні клієнтам технічної допомоги з програмним забезпеченням, мікрокомп'ютерами, безпекою та керування мережею. Ця допомога зазвичай надається через кол-центр. Персонал кол-центру TISSAT ділиться на три рівні:

- оператори першого рівня, які отримують запити клієнтів і відповідають на ті, з яких вони пройшли базову підготовку або на ті, чий рішення зареєстровано в записах компанії з протоколами дій;

- оператори другого рівня, які є технічними спеціалістами, відповідальними за вирішення проблем, які не змогли вирішити оператори першого рівня;

- провідні технічні спеціалісти та адміністратори, які відповідають за організацію робочих груп, призначення проблем конкретним спеціалістам і створення загальних рішень, які будуть зареєстровані та використані пізніше операторами нижчих рівнів.

Щоб гарантувати високу якість обслуговування, компанія підписує з компаніями, яким надає послуги, угоди про рівень обслуговування (SLA), де вказуються різні характеристики послуг, що надаються – описові мітки послуг, які ідентифікують кожен запит як належний до певного типу, пріоритети послуг, максимальний час допомоги та певні параметри, які вимірюють ступінь виконання цих послуг.

Після встановлення угод про рівень обслуговування компанія може узгодити надання послуг клієнтам за допомогою кількох каналів входу – телефонного дзвінка, веб-сайту, електронної пошти або чату. Коли центр обслуговування

отримує запит, створюється так званий реєстр інцидентів або заявка із даними клієнта та описом проблеми.

З точки зору клієнта, заявки характеризуються своїм станом (призначено, у процесі, вирішено, закрито, очікує на розгляд, вимагає розробки чи зміни програмного забезпечення), а також часом вирішення проблеми, що дозволяє клієнту знати ступінь виконання заявки.

Для центру обслуговування заявки характеризуються іншими параметрами, такими як категорія проблеми, до якого оператора або групи направлено заявку, робочі примітки про проблему, дані про програмне забезпечення та його дефекти. Процес вирішення проблеми генерує більше інформації, яка допомагає пояснити застосоване рішення – метод вирішення, рівень оператора, ключові слова, URL-адреси, вкладені документи або спостереження.

Щоб впоратися з цими проблемами, TISSAT розробила новий інструмент управління заявками служби підтримки I2TM (Intelligent and Integrated Ticketing Manager). Ця програма покращує роботу колишньої служби підтримки і, таким чином, якість підтримки клієнтів. Крім того, було проведено дослідження з метою прогнозування кількості майбутніх запитів на обслуговування, а також часу, коли вони відбудуться, щоб оптимізувати ресурси, що використовуються для вирішення цих подій. Одночасно була розроблена система, здатна пропонувати відповідні рішення для допомоги операторам. Ця система скоротила час вирішення проблем. Система управління заявками на основі випадків (Case-Based Reasoning for Ticketing Management (CBR-TM)) зберігає та повторно використовує інформацію про минулі проблеми та їх вирішення. CBR-TM була розроблена максимально гнучко, щоб полегшити його адаптацію до роботи з даними будь-якого нового проекту, яким компанія може керувати в майбутньому.

CBR-TM діє як окремий модуль I2TM для консультування щодо рішень. Обидві системи обмінюються даними та синхронізують їх за допомогою викликів веб-служб. Таким чином, їх архітектури є незалежними, і можна вносити зміни в одну з них, не впливаючи на іншу [10].

1.3 Постановка задачі дослідження

Проведений вище аналіз показав, що команди підтримки клієнтів ІТ-компаній відіграють чи не одну з найважливіших ролей у процесі замовлення та супроводу будь-якого ІТ-продукту чи послуги. Для правильної взаємодії служби підтримки з клієнтами, потрібно використовувати різні моделі управління, які дуже залежать від напрямку діяльності компанії у сфері ІТ. Модель управління повинна підбиратися відповідно до потреб окремої компанії. Вона повинна допомагати ефективно керувати наявними людськими ресурсами у службі підтримки для швидкого та якісного виконання запитів клієнтів.

Дослідження відомих підходів показало, що впродовж кількох останніх десятиліть найпопулярнішою моделлю є багаторівневий метод управління командою підтримки, оскільки даний підхід є максимально прозорий, зрозумілий та послідовний. Але із швидкими змінами в інформаційних технологіях, виникненням нових засобів та комп'ютерних систем для взаємодії з клієнтами та між членами команди, виникла потреба у переосмисленні багаторівневої моделі та створенні більш адаптивних, ефективних та гнучких методів для оптимізації процесу підтримки, скорочення часу вирішення проблем та збереження фінансових ресурсів компаній. Виникли нові підходи, які базуються на більш незалежній взаємодії команд підтримки, залучають усіх агентів до командної роботи та мають вищий рівень мотивації кожного окремого співробітника у плані особистого розвитку. Такі моделі управління можуть мати безліч варіацій та особливостей, більше того, можуть бути дешевшими в експлуатації та простішими у впровадженні.

Тому розробка та дослідження моделі управління командою підтримки виконання замовлень ІТ-компанії є актуальною задачею і метою даного дослідження.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- проаналізувати відомі моделі управління командами підтримки в ІТ-компаніях: традиційну багаторівневу модель та модель «роїння»;

- виділити переваги та недоліки обох моделей, а також труднощі пов'язані з їх імплементацією та використанням;
- розробити комбіновану модель управління командою підтримки виконання замовлень, яка ґрунтуватиметься на основі переваг обидвох моделей;
- здійснити математичний опис та на його основі порівняння багаторівневої моделі, моделі «роїння» та комбінованої;
- розробити нечітку систему оцінювання впливу моделі команди підтримки на виконання замовлень;
- дослідити запропоновану комбіновану модель на прикладі реальної компанії та проаналізувати отримані результати.

Висновки до розділу 1

1. Команди підтримки клієнтів ІТ-компаній відіграють чи не одну з найважливіших ролей у процесі замовлення та супроводу будь-якого ІТ-продукту чи послуги. Для взаємодії служби підтримки з клієнтами потрібно використовувати різні моделі управління, які повинні допомагати ефективно керувати наявними людськими ресурсами у службі підтримки для швидкого та якісного виконання запитів клієнтів.

2. Проаналізовано сучасні підходи до управління командами підтримки у ІТ-компаніях. Дослідження показало, що впродовж кількох останніх десятиліть, найпопулярнішою моделлю є багаторівневий метод управління командою підтримки, але в сучасних умовах виникла потреба у створенні більш адаптивних, ефективних та гнучких методів для оптимізації процесу підтримки, скорочення часу вирішення проблем та збереження фінансових ресурсів компанії.

2 МОДЕЛІ УПРАВЛІННЯ КОМАНДОЮ ПІДТРИМКИ ВИКОНАННЯ ЗАМОВЛЕНЬ ІТ-КОМПАНІЇ

2.1 Традиційна модель управління командою ІТ-підтримки

Технічна підтримка може складатися з одного співробітника для невеликих компаній або може включати багато відділів і працівників. Наприклад, велика ІТ-компанія може мати два типи команд:

- внутрішня команда ІТ-підтримки, яка допомагає співробітникам, коли вони мають справу з технічними проблемами;
- зовнішня команда підтримки допомагає клієнтам і користувачам послуг ІТ-компанії.

Технічна підтримка може надаватися різними способами, залежно від рівня підтримки, зокрема телефоном, електронною поштою, чатом або відео-зв'язком, чат-ботами, онлайн-підручниками та інструкціями, дошками оголошень та іншими інструментами реєстрації.

Структурування ІТ-підтримки за допомогою рівнів є зручним з кількох причин:

- стратегічне задоволення потреб клієнтів;
- створення позитивного клієнтського досвіду;
- швидке вирішення невеликих або простих у вирішенні проблем;
- встановлення термінів та протоколів для проблем, які складніше вирішити;
- підвищення задоволеності співробітників в команді;
- покращення навчання співробітників, підвищення мобільності та утримання;
- отримання відгуків і пропозицій щодо розвитку продукту.

Розглянемо багаторівневу модель управління командою підтримки, яка широко використовується в різних ІТ-компаніях.

Схема багаторівневої моделі управління командою ІТ-підтримки з коротко описаними ролями членів команд зображена на рисунку 2.1.

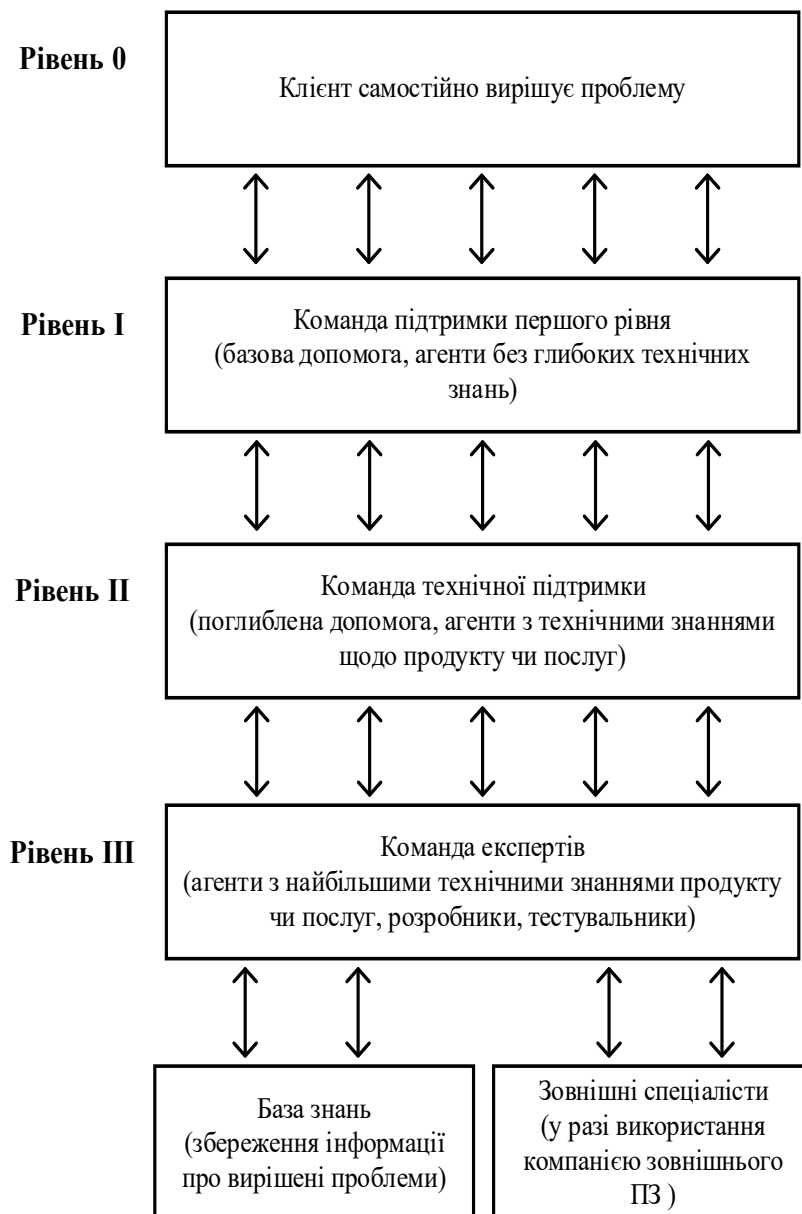


Рисунок 2.1 – Рівні IT-підтримки

На рисунку 2.2 зображений алгоритм роботи типової інфраструктури багаторівневої IT-підтримки.

Коли користувач чи клієнт звертається по допомогу, спочатку йому пропонують переглянути інформацію чи зробити замовлення самостійно, користуючись онлайн-ресурсами, які надає компанія. Це може бути веб-сторінка питань та відповідей, чат-бот чи форум. Всі ці дії відбуваються на рівні 0. У деяких випадках інформації достатньо, тому користувач залишає систему підтримки.

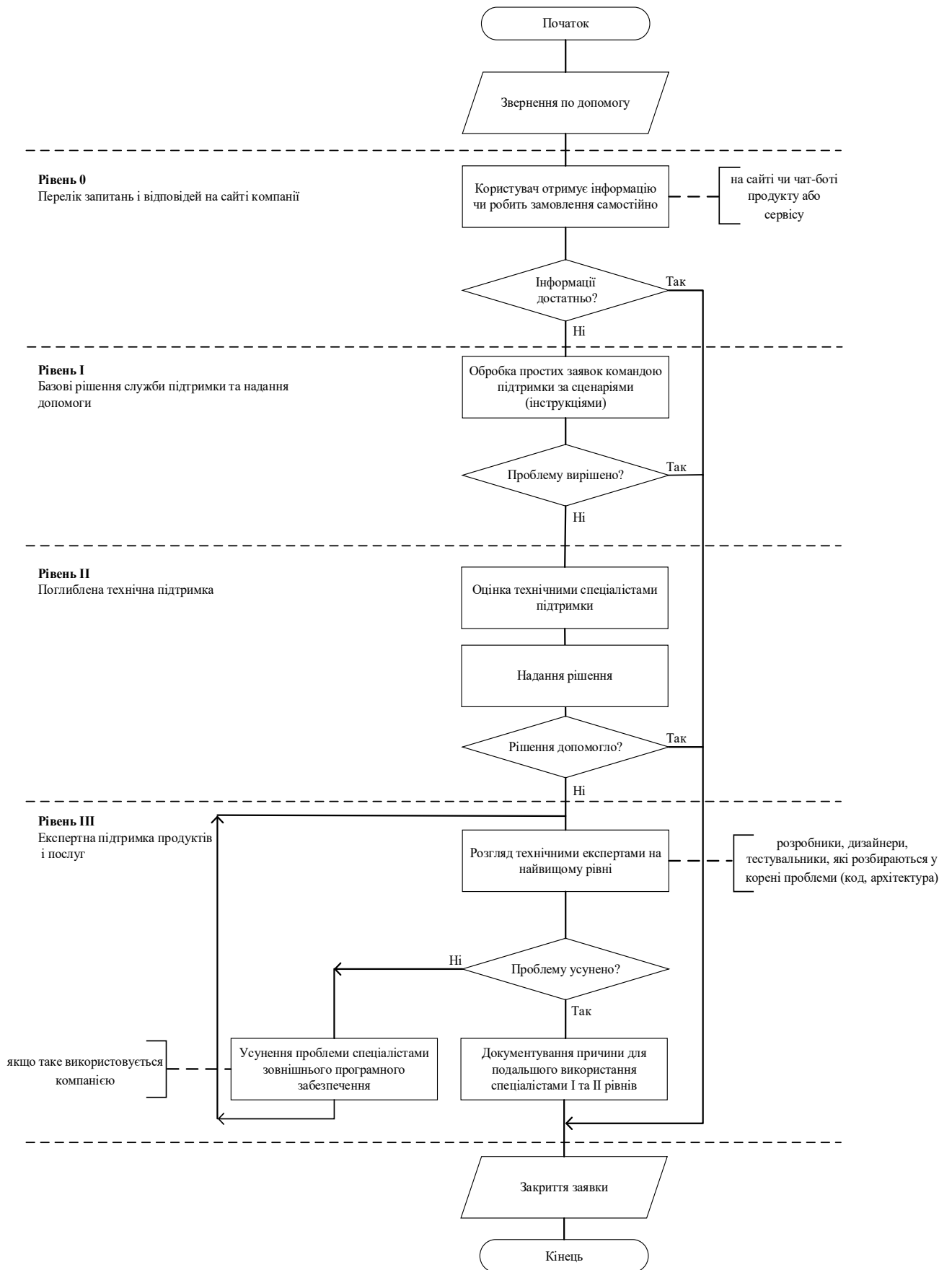


Рисунок 2.2 – Алгоритм багаторівневої ІТ-підтримки

У разі, якщо клієнту не допомагає наявна інформація, він звертається до першого рівня підтримки через певні канали зв'язку (чат, телефон, електронна пошта і тд.). На цьому етапі відбувається обробка запиту агентами першого рівня підтримки, зокрема визначається складність проблеми і, якщо агенту вдається допомогти, заявка закривається. Коли агент розуміє, що не може допомогти клієнту, його запит підіймається на рівень вище, де його розглядає спеціаліст технічної підтримки, з більш поглибленими технічними знаннями. Надається певне рішення проблеми і якщо воно допомагає, то заявка закривається.

У випадку, якщо рішення не працює, технічний спеціаліст направляє запит про проблему на найвищий рівень розгляду, де її аналізує команда технічних експертів. Це можуть бути співробітники, які безпосередньо брали участь у розробці продукту чи послуги, серед них є розробники, дизайнери, тестувальники, які, як ніхто, розбираються у продукті і можуть швидко вирішити проблему шляхом внесення змін у код, щоб запобігти її виникненню у майбутньому. У деяких випадках, коли компанія використовує зовнішнє програмне забезпечення і помилка стосується саме його, команда експертів звертається до розробників такого програмного забезпечення і вирішує цю проблему. Після цього відбувається документація проблеми та її рішення у базу знань для подальшого використання спеціалістами I та II рівнів. Врешті, заявка закривається і клієнта сповіщають про її успішне вирішення.

Незважаючи на те, що в основі традиційної моделі управління є лише три рівні, серед компаній, які досі використовують дану модель, часто зустрічається рівень 0 – це рівень самообслуговування. Зростаюча популярність порталів самопомоги у службі обслуговування користувачів може бути величезною для оптимізації ефективності команд підтримки. У цього рівня є наступні особливості:

а) методологія підтримки:

1) користувачі отримують інформацію про підтримку з веб-сторінок і програм для мобільних пристроїв, включаючи поширені запитання, детальну інформацію про продукт і технічну документацію, публікації в блогах, посібниках;

2) користувачі використовують програми для доступу до каталогів послуг, де вони можуть запитувати та отримувати послуги без залучення ІТ-персоналу;

3) електронна пошта, веб-форми та соціальні мережі, такі як Twitter, LinkedIn тощо, використовуються для надсилання запитань і запитів до вищих рівнів підтримки або персоналу компанії;

4) форуми дозволяють користувачам шукати спільні рішення, як правило, без участі персоналу компанії;

б) вимоги від персоналу:

1) рівень 0 вимагає технічних і маркетингових ресурсів для створення, підтримки та оновлення інформації про продукт;

2) команда розробників займається створенням веб-сайтів і програм підтримки;

3) модератори використовуються для моніторингу форумів користувачів та клієнтів;

4) персонал Рівня I відповідає на запити, отримані електронною поштою, веб-сайтами чи соціальними мережами.

Рівень I – служба обслуговування. Служба обслуговування – це перша лінія підтримки для користувачів, яка візьме на себе відповідальність за проблему та вирішить значний відсоток (зазвичай близько 70%) без ескалації на вищі рівні ІТ-підтримки. Служба обслуговування укомплектована фахівцями загального профілю, які мають широкі технічні та прикладні знання з обмеженою глибиною.

Рівень I повинен зібрати якомога більше інформації від клієнта. Зібрані дані можуть включати будь-що, починаючи від повідомлення про помилку, яке відображається на екрані, будь-яких файлів, журналів, знімків екрану, будь-яких використаних даних або послідовності кроків, виконаних кінцевим користувачем. Представники рівня I повинні мати можливість чітко визначити, чого клієнт намагається досягти, щоб переконатися, що правильні кроки буде вжито на наступних рівнях, якщо це необхідно. Виходячи із встановлених цільових часових рамок, їм надається обмежений час для вирішення більшості повідомлених

проблем. У цього рівня є наступні особливості:

а) методологія підтримки:

1) підтримка основних питань клієнтів, таких як вирішення проблем використання та виконання запитів служби підтримки, які потребують участі IT-фахівців;

2) якщо проблему неможливо вирішити, вони передають її до Рівня II;

б) вимоги від персоналу:

1) технічний персонал нижчого рівня підготовки, навчений вирішувати відомі проблеми та виконувати запити на обслуговування за допомогою сценаріїв.

Рівень II – технічні групи або групи керування програмами. Рівень II складається з технічних команд і груп керування програмами, у яких є люди з більш спеціальними знаннями та навичками, які можуть витратити більше часу на вирішення проблеми. Рівень II зазвичай вирішує близько 20% проблем, про які повідомляється, через їх вищий рівень знань. Вони можуть вирішувати складніші проблеми та мати доступ до фінансових платформ для повернення коштів. Після вирішення на рівні II служба обслуговування рівня I отримує сповіщення та зв'язується з користувачем, щоб закрити заявку підтримки. Якщо функції рівня II не можуть вирішити проблему, вони передають проблему до рівня III, групи, яка має ще більш поглиблені знання та навички для застосування до проблеми. У цього рівня є наступні особливості:

а) методологія підтримки:

1) досвідчені та обізнані технічні спеціалісти оцінюють запити та пропонують рішення для проблем, які не можуть бути вирішені на рівні I;

2) якщо рішення неможливе, служба підтримки рівня II передає інцидент до рівня III;

б) вимоги від персоналу:

1) допоміжний персонал із глибокими технічними знаннями продукту чи послуги, але це не інженери чи програмісти, які розробили та створили продукт.

Рівень III – підтримка розробників. Рівень III, третя лінія підтримки (зазвичай команда розробників), приймає будь-які вхідні ескалації від рівня II і визначає та призначає ресурси для вирішення проблеми. Після вирішення проблеми повідомляється рівень II, а також служба обслуговування рівня I, яка закриває заявку з клієнтом. Ресурси підтримки рівня III зарезервовані для вирішення лише найскладніших проблем, оскільки вони є дуже дорогим ресурсом. Заявка, яка досягає цього рівня підтримки, у багато разів дорожча, ніж заявка, вирішена на рівні I. Рівень III складається з інженерів підтримки, розробників та менеджерів з роботи з клієнтами, які мають доступ до найвищого рівня інформації про компанію/продукт. Рівень III зазвичай допомагає невеликому відсотку клієнтів, які або є цінними клієнтами, або клієнтами, які мають дуже незвичні технічні проблеми. У цього рівня є наступні особливості:

а) методологія підтримки:

1) доступ до найвищих доступних технічних ресурсів для вирішення проблем або створення нових функцій;

2) технічні спеціалісти рівня III намагаються відтворити проблеми та визначити їх першопричини, використовуючи дизайн, код або специфікації продукту;

3) після визначення причини компанія вирішує, чи створювати нове виправлення залежно від причини проблеми. Нові виправлення документуються для використання персоналом рівня I і рівня II;

б) вимоги від персоналу:

1) фахівці рівня III, як правило, є найбільш висококваліфікованими фахівцями з продукту, і можуть включати програмістів, головних архітекторів або інженерів, які створили продукт або послугу.

Також вирішення проблеми може покладатися на розробників зовнішнього програмного забезпечення, яке не розроблялося компанією. Це може бути більш поширеним у компаніях, які мають справу зі складнішими технологічними продуктами.

Багато компаній змінюють цей шаблон і комбінують рівні підтримки

відповідно до своїх ресурсів і філософії. У деяких організаціях функції рівня I і рівня II виконуються одним і тим же персоналом. Інші компанії можуть, наприклад, поєднувати функції рівня II і рівня III [11].

Інший спосіб розглядати рівні – це часовий показник: чим на вищий рівень переходить проблема, тим більше часу (і, можливо, ресурсів) знадобиться для її вирішення. В цьому випадку компанія може:

- встановити часові обмеження для рівнів. Наприклад, якщо проблема рівня I займає більше 15 хвилин, вона автоматично переходить до рівня II;

- дозволити персоналу IT-підтримки визначити, коли потрібно передати проблему на вищий рівень.

Тенденції, які зараз часто спостерігають у багаторівневій технічній підтримці, включають:

- а) автоматизація якомога більшої кількості технічної підтримки підвищує ефективність і доступність;

- б) самообслуговування та управління знаннями. Якщо інформація на рівні 0 легко доступна, клієнти можуть швидко знаходити рішення без допомоги IT-підтримки, заощаджуючи ресурси спеціалістів вищої кваліфікації для створення нових рішень і усунення складних проблем. Це поєднує дві практики: портали самообслуговування та управління знаннями.

- в) живі чати (без попереднього сценарію/інструкції). На вищих рівнях живі чати продовжують замінювати значну частину телефонних дзвінків. Якщо клієнтам запропонувати опцію живого чату, вони зможуть швидше вирішити своє питання, що може сприяти позитивному досвіду клієнтів.

Оскільки користувачі будь-якого продукту стають досвідченішими, агенти служби підтримки повинні не відставати – як щодо власного продукту, так і щодо продуктів конкурентів. З повним доступом до Інтернету знання є основою; сьогодні саме персоналізована допомога та обслуговування клієнтів стають засобом позитивного досвіду клієнтів [12].

Багаторівнева модель підтримки існує вже більше 30 років. Ця модель була прийнята у всьому світі тисячами організацій усіх типів і довела свою цінність.

Переваги та причини широкої популярності багаторівневої моделі підтримки:

- вона добре працює, коли великий відсоток проблем є відносно простими, такими, що їх може вирішити служба підтримки загального рівня I;

- багаторівнева підтримка також добре працює там, де немає великої частоти змін чи оновлень і розгортання нових версій у реальному середовищі. Коли середовище підтримки є відносно стабільним, лише з одним або двома основними оновленнями на рік і невеликими проміжними оновленнями функціональності, багаторівнева модель може працювати досить добре, швидко вирішуючи більшість проблем на рівні I і лише з невеликим відсотком проблем, які потребують рівня II або III;

- вона сумісна з традиційною структурою більшості ІТ-організацій: персонал організований у групи підтримки навколо технологій і програм;

- вона позиціонує спеціальну функцію як єдину точку контакту для користувачів, яка оптимізована для ефективної взаємодії з користувачем. Функція служби підтримки спеціалізується на навичках спілкування та управлінні взаємовідносинами з користувачами (залишаючи технічні функції та функції програми ізольованими від взаємодії з користувачами). Таким чином, спеціалізованим технічним і прикладним ресурсам не потрібно оптимізувати навички обслуговування клієнтів;

- знання та навички загального рівня підтримки дешевші та легші для розгортання на рівні I, ніж більш спеціалізовані технічні та прикладні ресурси;

- вона добре задокументована і відносно проста у реалізації, оскільки структура служби підтримки та інші функції підтримки були включені в офіційні публікації передового досвіду;

- навчальні матеріали та курси також існують вже протягом багатьох років, оснащуючи рівень I та II знаннями про їхні ролі та забезпечуючи навчання відповідним соціально-комунікативним і технічним навичкам.

Зважаючи на ці фактори, існуючим користувачам, а також тим, хто розглядає багаторівневу підтримку для використання в компанії, зрозуміло, що вона загалом працює та забезпечує життєздатний підхід до вирішення проблем, про які

повідомляють користувачі. Однак бізнес і технології змінюються. Прихильники багаторівневої моделі підтримки вже стикаються з дедалі більшими проблемами [13].

Швидкість змін у бізнесі та технологіях призвела до зростаючого попиту на безперервне розгортання нових функцій і покращення ІТ-послуг у багатьох організаціях. Це призводить до більш частого внесення змін і оновлень у реальне середовище, що загалом впливає на стабільність і викликає вищий рівень подій та інцидентів, на які команда підтримки повинна швидко реагувати та вирішувати. Персонал технічної підтримки рівня I у таких центрах підтримки має проблеми зі збільшенням кількості заявок через більший обсяг частих змін і оновлень.

Фактором, який спонукає до змін у моделі підтримки, є все більше впровадження самообслуговування. Організації, які досягли успіху в розгортанні самообслуговування, надають користувачам знання для вирішення вищого відсотка простіших проблем самостійно, залишаючи складніші проблеми службі підтримки. Якщо вони не розгорнули успішний процес управління знаннями разом зі зрілими процесами та хорошими робочими стосунками між командами підтримки, такі центри підтримки часто виявляються погано обладнаними для вирішення зростаючого відсотка складних проблем. Як це не дивно, але ефективно самообслуговування може призвести до збільшення кількості проблем, переданих до технічних і прикладних груп рівня II і III, а не до зменшення кількості проблем.

Багаторівнева модель підтримки є, по суті, лінійною за своєю природою, коли один рівень передає проблеми на наступний рівень, сподіваючись, що вони зможуть вирішити проблему. Через свою структуру, опору на окремі функціональні групи та процес ескалації багаторівнева модель підтримки не допомагає організаціям ефективно вирішувати ці проблеми. Варто розглянути недоліки багаторівневої моделі, а саме:

а) вона не заохочує співпрацю між командами та обмін знаннями. Багаторівнева модель припускає, що ІТ-компанія організована навколо окремих самостійних одиниць: служби обслуговування, груп технологій і програм, а також ІТ-операцій. Якщо не впроваджено чіткі політики взаємодії, а також супровідні

угоди між рівнями підтримки та якісне управління знаннями, то змусити групи підтримки ефективно працювати разом для вирішення складних запитів клієнтів із цією моделлю дуже проблематично. Знання виникають і мають тенденцію залишатися в спеціалізованих групах, якщо не вживаються заходи для обміну досвідом між групами. Персонал першого рівня є ізольований від інших рівнів і не завжди має можливість співпрацювати з іншими підрозділами, щоб розвивати свої навички та знання;

б) модель припускає неправильну класифікацію складних проблем на рівні I. Через брак спеціальних знань і досить часто через недосконалу схему та процес категоризації підтримка рівня I не завжди ефективна для класифікації інцидентів і запитів. Ефективне управління знаннями та ефективна схема класифікації можуть допомогти усунути цей недолік. Інакше, неправильна класифікація через брак знань може призвести до неправильної ескалації та перекидання справи між рівнями без ефективного рішення. Це призводить до збільшення середнього часу вирішення, більшого незадоволення користувачів і більшої середньої вартості вирішення проблеми;

в) самообслуговування призводить до того, що більший відсоток проблем є складними. У міру того як самообслуговування продовжує набувати все більшого поширення, дедалі більша кількість простіших проблем вирішується кінцевим користувачем на рівні 0. Служба обслуговування стикається зі зростаючою кількістю складних проблем, які вимагають більш глибоких знань і досвіду на рівні I. Проте багаторівнева модель підтримки не сприяє неперервній передачі знань, яка потрібна. Як наслідок, багаторівнева модель з часом викликає більше ескалацій;

г) тенденція до постійного розгортання нових функцій. Багаторівнева модель підтримки не містить чітких вказівок для груп технічних спеціалістів і спеціалістів із застосування першого рівня; замість цього модель вимагає ескалації та фільтрації проблем до рівня II і III;

д) багаторівнева підтримка призводить до черг, незавершеної роботи та відставання. При багаторівневій підтримці черги встановлюються для кожного

спеціаліста рівня II і III, щоб підтримувати передачу проблем цим групам. Результатом цього є те, що певна кількість питань нерозглянута, очікуючи на спеціалістів, яким також доручено розробити та перенести оновлення. Незавершена робота створює затримку, збільшуючи середній час вирішення проблеми, особливо тому, що більший відсоток питань стає складнішим. Робота в черзі та відставання також збільшують фінансові витрати, оскільки задіяно кілька сторін і потрібно більше часу для вирішення проблеми [14].

У зв'язку із переходом до менших портативних програм, зростанням конкуренції та зростанням тенденції розвитку нових технологій все більше і більше компаній шукають своїх постачальників ІТ-послуг для впровадження більш частих функціональних оновлень і оновлень продуктивності, що додають цінності кінцевому ІТ-продукту, з незначною кількістю дефектів та за доступною ціною. Щоб і надалі залишатися релевантним і цінним діловим партнером, ІТ-компанії повинні реагувати на зміни в бізнес-тенденціях, а також у бізнесі клієнтів, яких вони обслуговують. Багаторівнева модель підтримки має багато переваг, але має і деякі труднощі. Щоб справлятися зі зростаючою складністю проблем, доцільно також дослідити інші моделі підтримки, які заохочують співпрацю та обмін знаннями між командами підтримки, забезпечують швидке реагування та вирішення проблем і підтримують високу цінність усіх команд.

2.2 Модель «роїння»

У деяких випадках багаторівнева модель підтримки більше не здається оптимальним підходом для середовищ підтримки. В організаціях, які зазнають частих і швидких змін у бізнесі, клієнти вимагають постійного розгортання нових функцій і негайного звернення уваги на будь-які збої або погіршення продуктивності. У цих випадках багаторівнева підтримка недостатня. Модель роїння з її акцентом на обробці проблем у реальному часі може забезпечити швидші та ефективніші відповіді та рішення.

Зростаюча популярність і впровадження методів гнучкої (Agile) розробки та

моделі співпраці DevOps групами розробки та операцій добре поєднується з моделлю підтримки роїння, оскільки вона спеціально спрямована на необхідність подолання організаційних проблем і заохочує співпрацю.

Дедалі більше впровадження самообслуговування призводить до збільшення частоти простіших проблем, які вирішуються користувачами, і більш складних проблем, які залишаються для вирішення службою підтримки. Модель підтримки роїння пропонує більш ефективні робочі групи, які можуть швидко та якісно вирішувати дедалі складніші проблеми, які повідомляються користувачами.

Багаторівнева модель підтримки за своєю природою збільшує витрати, коли проблема переходить із рівня I до рівня II та рівня III. Вирішення проблем під час першого контакту або, принаймні, у центрі підтримки за допомогою ефективних робочих груп може допомогти задовольнити потреби організацій зменшити витрати, підвищивши продуктивність.

Модель підтримки «роїння» слід розглядати та використовувати, коли це має зміст з огляду на середовище підтримки та IT-компанії. Насамперед ця модель ефективна в середовищах, де швидкі зміни в бізнесі призвели до впровадження практик Agile та DevOps, або коли поточна оцінка стану організації підтримки показує низьку продуктивність у вирішенні серйозних проблем, відносно довгий час вирішення, порівняно із середніми показниками по галузі та низький рівень першого контакту з високим відсотком проблем, переданих до груп спеціалістів з технічної підтримки.

У моделі «роїння» команда підтримки створює можливості та потенціал, одночасно оптимізуючи здатність людей робити спільний внесок; підвищує залученість і лояльність; покращує успіх і цінність клієнта завдяки вдосконаленому вирішенню проблем шляхом збільшення охоплення та релевантності.

Модель «роїння» дуже відрізняється від багаторівневого підходу, адже вона заснована на співпраці, а не на ескалації. У моделі «роїння» член команди підтримки обробляє заявку від початку до кінця замість того, щоб «проштовхувати» її через багаторівневу модель.

У роївій моделі член команди, який швидше за все зможе обробити заявку,

отримує завдання вирішити цю заявку. Коли користувачеві потрібна допомога, замість того, щоб передавати заявку між командами, команда підтримки «роїться» навколо проблеми, знаходячи відповідь, доки заявка не буде закрито [15].

На рисунку 2.3 подана схема вирішення проблеми за допомогою моделі підтримки «роїння» або співпраці.

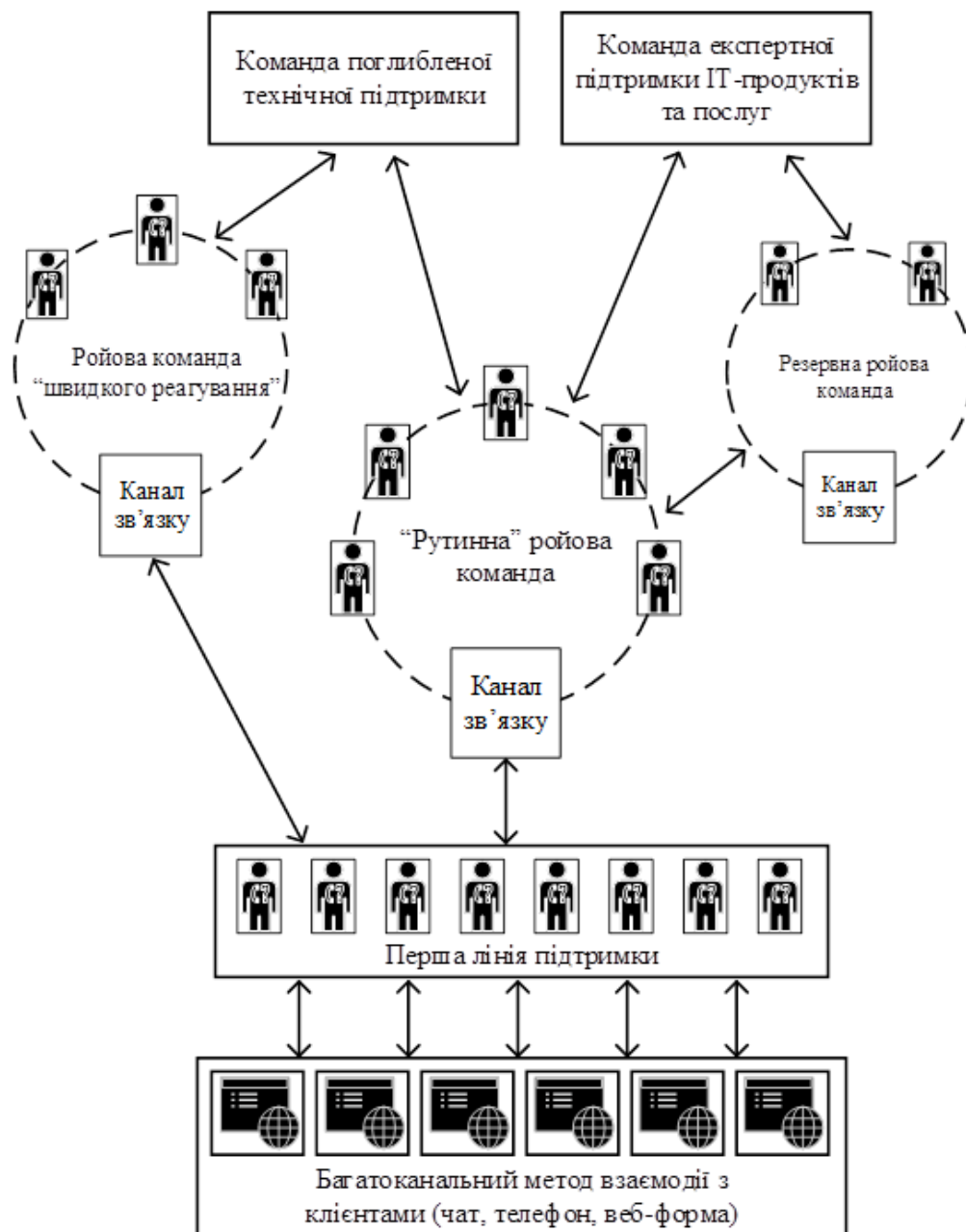


Рисунок 2.3 – Схема моделі «роїння» ІТ-підтримки

Для досягнення цілей цього сценарію, заздалегідь створюється три ройові групи з експертів та спеціалістів з технічних питань, які регулярно чергуються, щоб допомогти укомплектувати ці ройові команди:

- а) основна «рутинна» ройова команда, яка вирішуватиме більшість проблем;
- б) ройова команда «швидкого реагування», яка обробляє лише найпріоритетніші проблеми;
- в) «резервна» ройова команда, яка буде активно вирішувати будь-які питання, які дуже складні або не були вирішені раніше.

На рисунку 2.4 представлений алгоритм ІТ-підтримки згідно моделі «роїння».

Спочатку клієнт звертається із запитом та він опрацьовується першою лінією підтримки, де класифікується за рівнем пріоритетності.

Проблеми із високим пріоритетом відправляються до команди «швидкого реагування», яка аналізує проблему та працює з командами технічної та експертної підтримки до повного вирішення проблеми в найкоротші терміни.

Проблеми зі звичайним рівнем пріоритетності відправляються на опрацювання «рутинною» командою, яка також співпрацює з командами технічної та експертної підтримки.

Якщо заявку не вдається вирішити за певний проміжок часу, вона відправляється на обробку «резервною» командою.

Ця команда супроводжує цю заявку до її повного вирішення.

Незалежно від того, якою командою опрацьовувались заявки, після їх вирішення всі дані та рішення щодо проблем зберігаються у базі знань служби підтримки.

Ця інформація опрацьовується командою першої лінії підтримки, заявка закривається, а клієнта повідомляють про успішне вирішення його проблеми.

Опис робочого процесу з використанням моделі «роїння»:

- заявка на послугу передається в першу лінію підтримки через доступні багатоканальні методи взаємодії (чати, дзвінки, веб-форми). Користувач або клієнт надає облікові дані та те, з чим він звертається до центру підтримки, встановлюючи у системі статус абонента, будь-які спеціальні дії, які можуть знадобитися, та тип проблеми;

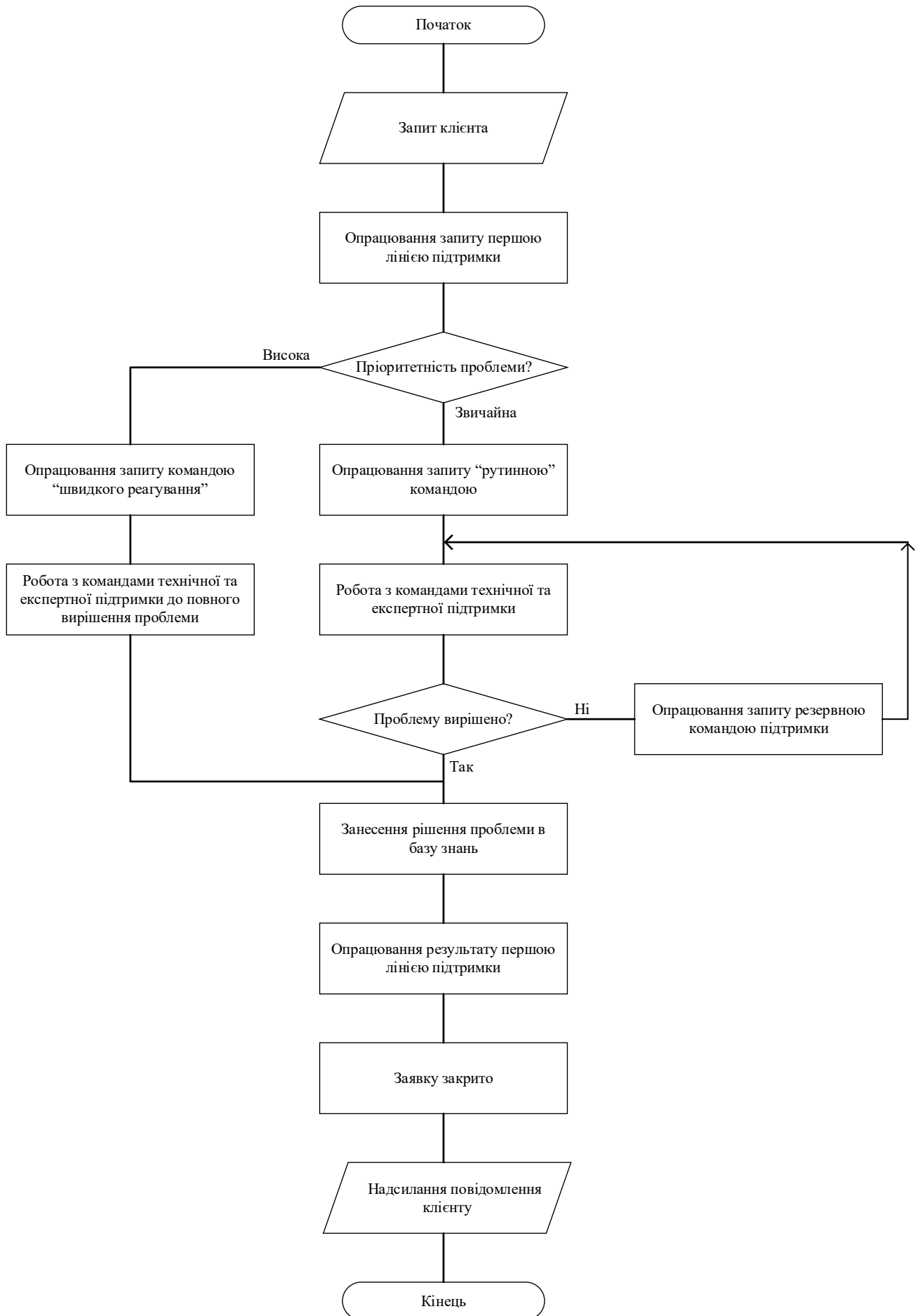


Рисунок 2.4 – Алгоритм ІТ-підтримки роїння

- перша лінія підтримки, використовуючи профілі та інформацію, надану клієнтом, визначає команду або конкретну особу, яка має найкращі знання та навички для вирішення цієї проблеми. Якщо проблема має високий пріоритет, вона негайно перенаправляється до групи «швидкого реагування» для вирішення. Команда «швидкого реагування» вирішує проблему або працює спільно з командою експертної підтримки, яка безпосередньо займалася розробкою сервісу чи продукту. Співпраця ведеться в режимі реального часу, доки проблему не буде вирішено;

- якщо проблема не є високопріоритетною, вона перенаправляється до «рутинної» команди або до особи, яка найкраще вирішить цю проблему. Відбувається спроба вирішити проблему при першому контакті з клієнтом, член команди використовує свої відповідні навички, знання та доступ до інтегрованої бази знань, щоб вирішити проблему або ж звертається до спеціалістів та експертів технічної підтримки. Якщо рішення знаходять, вони надаються члену команди підтримки для застосування, щоб вирішити проблему з клієнтом. Інші члени команди, пов'язані з проблемою, одразу бачать проблему, завдяки синхронізації заявки про проблему з їхніми профілями;

- якщо агент «рутинної» команди не може вирішити проблему, він сигналізує відповідним членам групи про те, що йому потрібна допомога. Це можна зробити кількома способами, наприклад, за допомогою інструменту чату, або опублікувати проблему на спільній дошці обговорень для менш термінових питань. Відповідні члени команди погоджуються та надають допомогу, співпрацюючи над вирішенням проблеми та прискорюючи її вирішення;

- «рутинна» команда підтримки також збирається та переглядає ще не вирішені заявки кілька разів на день. Інструменти співпраці, такі як чат і дискусійні форуми, використовуються для мозкового штурму всередині групи та з іншими групами спеціалістів для визначення можливого рішення. Вони співпрацюють, щоб вирішити проблеми, запобігаючи якомога більшій ескалації на інші рівні. Якщо проблему неможливо вирішити, вони проводять валідацію проблеми у системі (перевіряють чи проблема дійсно існує), перш ніж передавати її до відповідальної

групи технічної та експертної підтримки;

- високий відсоток проблем вирішується під час першого контакту (або принаймні в перший день) «рутинною» командою підтримки. Проблеми з високим пріоритетом обробляються в режимі реального часу паралельно командою «швидкого реагування», доки вони не будуть вирішені. Будь-які проблеми, що залишилися після цього, вирішує третій тип «резервної» ройової команди;

- «резервна» команда зосереджується на питаннях, які ще не вирішені, і щодня працює над вирішенням цих проблем. Це звільняє технічні команди та команди експертів з програм та послуг (програмістів, інженерів, дизайнерів) від необхідності безпосередньо мати справу з ескалацією заявок і гарантує, що міжфункціональна команда відповідних технічних спеціалістів підтримки своєчасно вирішить проблему [16].

Враховуючи середовище, належне планування та реалізацію, модель підтримки «роїння» може забезпечити такі результати:

а) більше відчуття спільного бачення, місії та мети в усіх групах підтримки, які є частиною постачальника ІТ-послуг, оскільки люди більше ідентифікують себе не за членством в організаційних або технічних командах, а за спільними цінностями, спільною метою та галузі знань. Вимірювання та показники команди мають пріоритет над індивідуальними показниками та звітністю. У цій моделі загальний набір цілей та завдань продуктивності розділяються між усіма членами команди;

б) вища задоволеність користувачів завдяки швидшому середньому часу виконання заявок. Це призводить до кращого утримання клієнтів і збільшення продажів продуктів чи послуг. Витрати також зменшуються – нижча середня вартість одної заявки завдяки меншій ескалації та швидшому середньому часу виконання;

в) вищий рівень задоволеності допоміжного персоналу, завдяки більшому обміну знаннями, що є невід'ємною частиною моделі роїння, і меншій рутинній роботі першої лінії підтримки.

г) задоволеність персоналу та підвищення морального стану, і як наслідок,

плинність кадрів зменшується, а утримання співробітників і продуктивність зростають.

Однією з кращих переваг моделі «роїння» є прозорість процесу, клієнт безпосередньо спілкується з агентом, який виконає його запит з початку до кінця. Багаторівнева модель підтримки часто тримає працівників і клієнтів у незнанні, оскільки агент підтримки, який передає заявку до наступного рівня, ніколи не дізнається про вирішення цієї заявки. У моделі «роїння» клієнти впевнені, що їхнє звернення буде розглянуто підходящим фахівцем із команди підтримки, здатним відповісти на заявку самостійно.

Команди підтримки змушені проявляти ініціативу та гуртувати членів команд розробників, щоб знайти відповідь на проблему. Таким чином, команда підтримки передає запитання клієнта напряду розробникам, щоб вони розуміли, що найкраще підходить для клієнтів компанії.

Крім того, у багаторівневій моделі працівники служби підтримки рівня I зазвичай проходять дуже мало навчання та повинні дотримуватися сценаріїв (наперед вивчені інструкції) для розгляду заявок. Перебуваючи на нескінченній черзі дзвінків та чатів, ці агенти, як правило, швидко змінюють роботу. Модель «роїння» має тенденцію підтримувати мотивацію співробітників, оскільки вони мають можливість вивчати нові навички та розвивати свій досвід [17].

Модель «роїння» також є гнучкою, легко адаптується та може бути реалізована кількома способами залежно від організації та середовища підтримки. Тим не менш, є деякі труднощі, які необхідно вирішити, щоб ефективно спланувати та реалізувати таку модель:

- налаштування профілів персоналу є абсолютно необхідним для ефективного вирішення вхідних проблем. Для кожного члена команди встановлюються задокументовані, постійно підтримувані профілі знань, загальних і спеціальних навичок, які зберігаються в системі керування послугами та сприяють направленню проблеми до найкращої особи чи групи для вирішення проблеми клієнта;

- процес управління знаннями та допоміжна база знань мають бути

інтегровані в процес обробки проблем, щоб вирішення проблем або повторно використовувалися, або фіксувалися в робочому процесі для того щоб не виконувати ті ж самі кроки повторно при подібній проблемі;

- ефективна категоризація та схеми пріоритетів гарантують належне визначення проблеми;

- спільна допоміжна система керування послугами для управління проблемами разом із ефективними інструментами співпраці. Проблеми все одно потрібно ідентифікувати, реєструвати та відстежувати протягом усього життєвого циклу. Інструменти для співпраці, які підтримують чат у режимі реального часу та інтегруються з іншими системами та соціальними мережами, забезпечать підтримку співпраці між групами в режимі реального часу;

- планування та створення ройових команд. Кожен «рій» насправді є невеликою командою, яка в режимі реального часу зосереджена на вхідних проблемах залежно від природи та пріоритету проблеми. Немає встановленого правила щодо того, як ви повинні організувати команди підтримки роїв, але одним із прикладів може бути рій для звичайних проблем, інший – для проблем високого пріоритету, а інший – для вирішення невиконаних завдань;

- доцільно визначити порядок ротації членів із технічної та експертної груп підтримки. Вона має відбуватися щотижня або щомісяця, щоб мінімізувати переривання проектних заходів.

Модель «роїння» працює лише за значної підтримки керівництва компанії, оскільки вимагає послаблення правил, які часто вкорінені в практиках, показниках і стимулах. Це вимагає від менеджерів відмовитися від традиційної моделі керування, щоб забезпечити більш адаптивну культуру самозабезпечення серед своїх співробітників.

Технічна підтримка протягом тривалого періоду розробила численні соціальні конструкції, які по-різному несумісні з переходом до більш спільного середовища роїння. Наприклад, домовленості про робочий рівень, що стосуються конкретної команди, перешкоджають міжкомандній співпраці. Проблемою є управлінська ієрархія: групи можуть бути зацікавленими сторонами в одній

проблемі, але якщо вони звітують через різну вертикаль в управлінській ієрархії, вони можуть мати суперечливі цілі, що перешкоджає співпраці.

Результатом цих проблем є те, що роїння може сприйматися як ризиковане та складне, і може розглядатися як щось, що неможливо легко реалізувати без складних і, можливо, дорогих організаційних змін.

Підприємства, які вивчають модель «роїння», як потенційну модель до впровадження, дуже часто висловлюють занепокоєння щодо відчутного збільшення витрат при її імплементації та експлуатації.

У багаторівневому середовищі підтримки зазвичай передбачається, що персонал рівня I буде дешевшим, ніж більш досвідчений, більш спеціалізований технічний персонал на рівні III. Часто існує думка, що модель «роїння» залучає дорогий персонал до процесу на більш тривалий період часу, частіше, ніж у традиційній багаторівневій моделі, і тому, середня вартість вирішення кожної проблеми буде вищою. Контраргументом до цієї думки є те, що модель «роїння» забезпечує довгострокову вигоду: через загальне скорочення роботи служби підтримки, покращену надійність обслуговування та завдяки кращій довгостроковій можливості першої лінії підтримки та «рутинної» команди вирішувати майбутні проблеми без необхідності залучати спеціалістів з технічних команд підтримки. Але даний контраргумент буде важко довести без належного практичного застосування моделі.

Коли підтримка організована навколо чітко визначених задач, часових рамок для ключових подій (таких як отримання та подальша передача частини роботи) можна поєднувати та усереднювати ці показники для того, щоб забезпечити узгоджене порівняльне вимірювання швидкості реагування підтримки. Ефективність цих показників зведена до нуля через модель «роїння».

Ця проблема впливає на вимірювання окремих осіб і команд, а також третіх сторін, таких як постачальники послуг і підрядники. Не так просто оцінити внесок будь-якої конкретної особи чи команди в більш вільно організовану модель співпраці [18].

Для багатьох організацій це більше, ніж просто вимірювання – це фінансове

завдання. Контракти про надання послуг часто містять заохочення, бонуси та штрафи, підкріплені цими традиційними показниками. В середовищах, для яких такі показники вкорінені, перехід до нової практики роботи вимагає не лише культурних змін, це потребує комплексного розгортання та заміни існуючих контрактних угод працівників.

У складній організації може бути дуже важко зрозуміти, хто є хто. Особі, яка намагається скликати рій, зазвичай потрібно отримати інформацію від кількох предметних областей. Переваги «роїння» в продуктивності зменшуються, якщо агенти витрачають багато часу просто на пошук того, хто міг би їм допомогти.

Деякі компанії вирішують цю проблему за допомогою підходу «інтелектуального роїння». Ця концепція спирається на профілі людей і моделі «репутації». Профілі в цій моделі містять інформацію про навички кожної особи: технічні чи «м'які» (взаємодія з людьми). Репутація створюється з часом роботи певної людини, шляхом моніторингу та вимірювання цінності, яку додав агент під час попередньої роботи у відповідних предметних областях. Поширеною проблемою щодо цього, є ризик надмірного навантаження на високоефективних людей надмірними запрошеннями до роїння навколо певної проблеми: важко збалансувати бажання досягти оптимальних результатів шляхом вибору найкращих людей із ризиком «виснажити» цих людей через надмірну залежність від них.

2.3 Комбінована модель ІТ-підтримки

У якості пропонованої моделі виступає дещо допрацьована та розширена модель управління командою ІТ-підтримки «роїння». Тобто за основу взято архітектуру моделі «роїння» з елементами багаторівневої моделі. Це дозволить використовувати переваги обох моделей та ефективніше вирішувати завдання поставлені перед відділом ІТ-підтримки компанії.

На рисунку 2.5 зображена схема вирішення проблеми за допомогою комбінованої моделі команди підтримки ІТ-компанії.

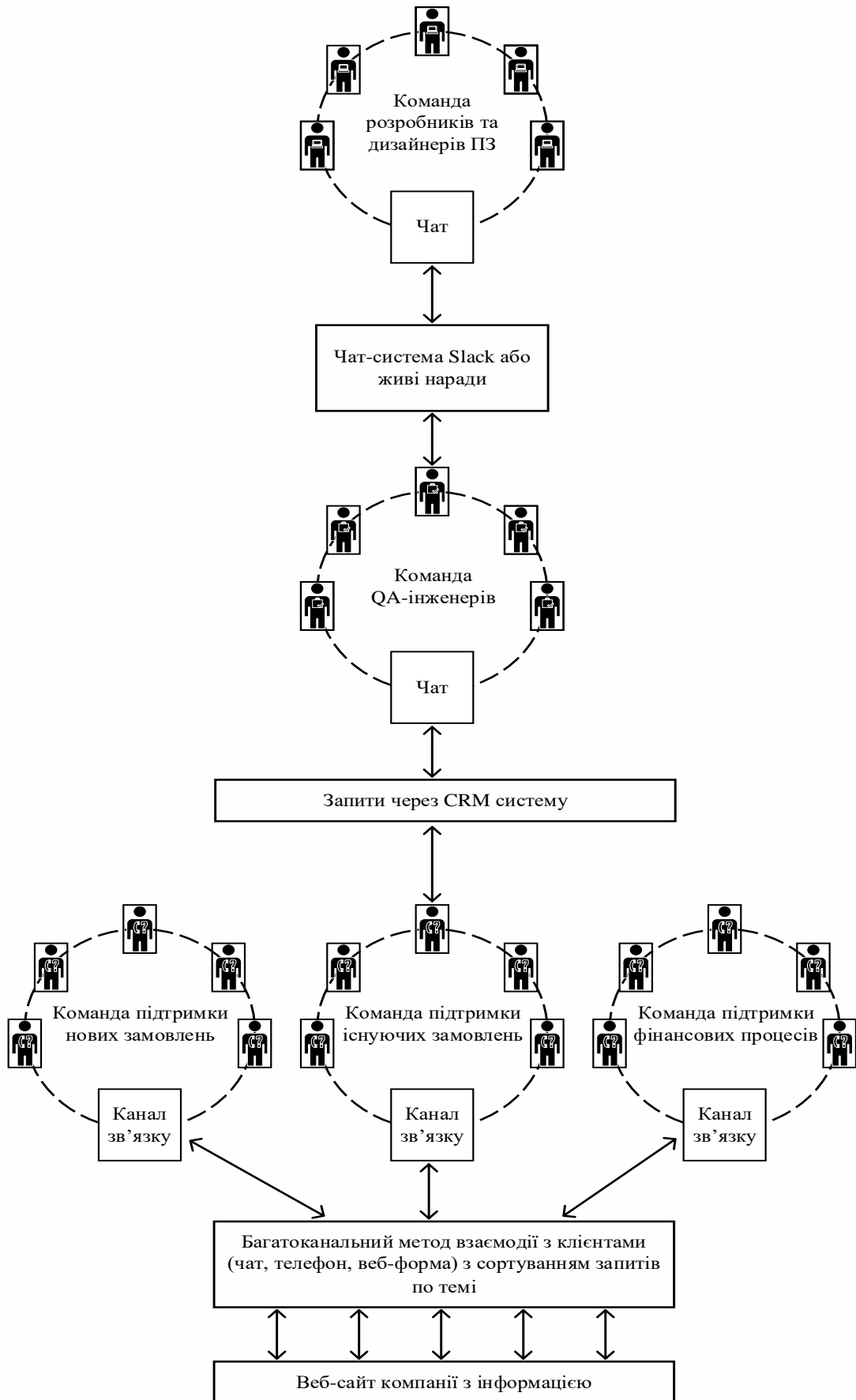


Рисунок 2.5 – Схема комбінованої моделі ІТ-підтримки

Для правильного функціонування цієї моделі заздалегідь потрібно сформуванню кілька команд спеціалістів підтримки та команд технічних спеціалістів. Члени кожної сформованої команди будуть співпрацювати один з одним в середині кожної команди для досягнення рішення проблеми якомога ефективнішим та швидшим шляхом. Доцільно виділити такі команди:

а) команда підтримки нових замовлень займається обслуговуванням нових клієнтів компанії, які хочуть вперше замовити програмний продукт чи послугу та потребують певної допомоги із замовленням;

б) команда підтримки існуючих замовлень займається обслуговуванням клієнтів та підтримкою їх продуктів, сервісів та послуг, які були створені компанією раніше. Також командою приймаються запити на зміну існуючих функцій, або ж запити на додавання нового функціоналу у продукт;

в) команда підтримки фінансових процесів допомагає клієнтам та користувачам, у яких виникли проблеми при оплаті за замовлення, або тим, які мають проблеми з поверненням коштів;

г) команда QA-інженерів – спеціалісти, які відповідають за якість продуктів, послуг та сервісів при їх впровадженні та експлуатації. Вони мають глибокі знання у продуктах компанії і можуть надавати рішення проблеми у разі, якщо вона не потребує виправлення у коді або функціоналі продукту. Або надають поради, як вирішити ту чи іншу проблему, поки помилку не виправили програмісти;

д) команда розробників та дизайнерів ПЗ – це найбільш обізнані співробітники компанії, які вузько спеціалізуються на конкретному продукті, послугі чи сервісі.

Клієнти взаємодіють з трьома типами команд першого рівня через визначені канали зв'язку відповідно до типу їх запиту (існуючі, нові замовлення, фінансові питання).

Після цього відповідна команда «роїться» навколо проблеми та у разі неможливості її вирішення, шляхом створення запитів в системі керування клієнтами (CRM), відправляє проблему до команди QA-інженерів.

Інженери забезпечення якості у свою чергу перевіряють отриманий запит, обговорюють цю проблему у команді та приходять до спільного вирішення проблеми.

У разі, якщо їм не вдається вирішити проблему, залучаються вже розробники та дизайнери конкретного продукту, на який прийшла заявка про проблему.

Розробники на своєму рівні обговорюють цю помилку та надають їй остаточне рішення, виправляють її та заносять інформацію у базу знань.

На рисунку 2.6 представлений алгоритм комбінованої моделі ІТ-підтримки.

Спочатку клієнт звертається за допомогою до веб-сайту компанії, де є уся наявна інформація про програмні продукти, сервіси та послуги, які надає ІТ-компанія.

На цьому веб-сайті детально описано, як робити замовлення, інформація про оплату за сервіси та є секція з часто вживаними питаннями.

У разі, якщо користувач не знаходить потрібних йому відповідей, він має можливість звернутись до автоматичного чату, у якому йому пропонується зробити вибір з трьох можливих тем: нове замовлення, існуюче замовлення, або ж фінансові питання.

Після того, як клієнт обирає потрібну йому тему запиту, його направляють у відповідну команду підтримки, ця команда тісно працює з клієнтом і намагається вирішити його питання.

Коли команда підтримки не в змозі вирішити проблему користувача, вона звертається до інженерів якості через автоматизовану систему керування клієнтами.

Інженери якості працюють над проблемою та надають потрібне рішення, або починають працювати з розробниками продукту.

Хоч вони і контактують з ними, розробники на своєму рівні «рються» навколо своєї проблеми без участі QA-інженерів доки не вирішать проблему.

Коли проблема отримує якесь рішення, вона документується та заноситься у базу даних компанії. Заявка клієнта закривається.

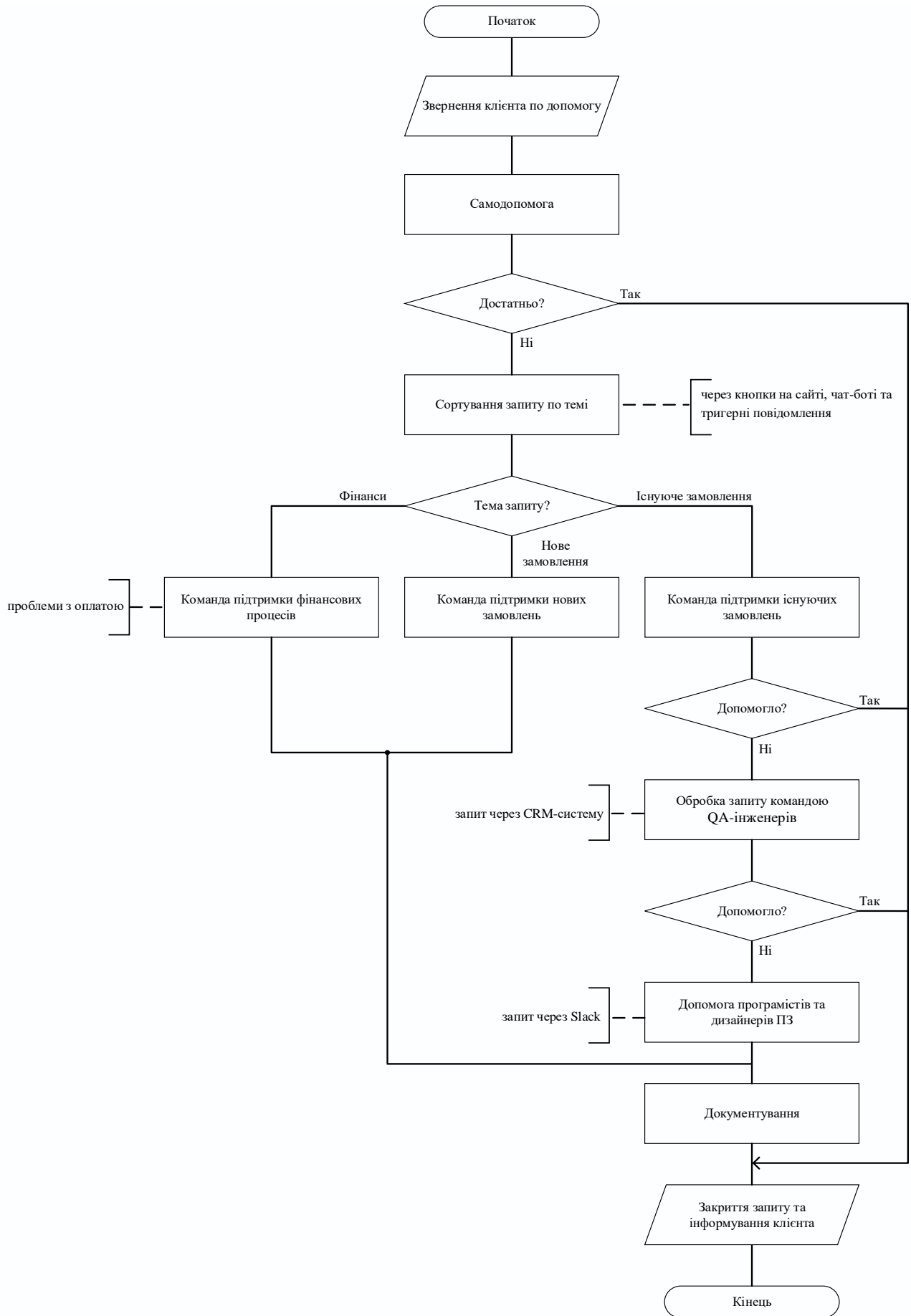


Рисунок 2.6 – Алгоритм комбінованої моделі IT-підтримки

Висновки до розділу 2

1. Проведено дослідження багаторівневої моделі управління командами ІТ-підтримки та моделі «роїння».
2. Розроблено й описано схеми та алгоритми діяльності команди підтримки відповідно до цих моделей. Розглянуто переваги та недоліки кожної з моделей, приведені приклади труднощів, які можуть виникнути при їх імплементації.
3. На основі проведеного аналізу багаторівневої та моделі «роїння» запропоновано комбіновану модель управління ІТ-підтримкою, в якій за основу взято архітектуру моделі «роїння» з елементами багаторівневої моделі. Це дозволить використовувати переваги обох моделей та ефективніше вирішувати завдання, поставлені перед відділом ІТ-підтримки компанії.
4. Розроблено схему та алгоритм комбінованої моделі підтримки.

3 ДОСЛІДЖЕННЯ КОМБІНОВАНОЇ МОДЕЛІ ІТ-ПІДТРИМКИ

3.1 Дослідження на основі математичного моделювання

Нехай, в ІТ-компанії сформовано команду підтримки із n виконавців, які б могли виконувати m функцій, необхідних для виконання запитів і замовлень клієнтів. Тоді загальний склад команди (HR) буде становити множину виконавців

$$HR = \{h_1, h_2, \dots, h_n\}, \quad (3.1)$$

а виконувані ними роботи проекту будуть пов'язані з кількістю функцій у вигляді множини

$$W = \{w_1, w_2, \dots, w_m\}. \quad (3.2)$$

3.1.1 Багаторівнева модель

Склад потенційних членів команди багаторівневої підтримки може бути описаний у вигляді формули:

$$HR = \{B_n, S_n, E_n, V_n\}, \quad (3.3)$$

де HR - загальний склад членів команди підтримки;

B_n – група базової допомоги;

S_n – група технічної підтримки;

E_n – група експертів;

V_n – зовнішні спеціалісти.

Клієнт K при виникненні проблеми P , маючи доступ до інформації, спеціально сформованої для таких випадків ІТ-компанією та відкритої для ознайомлення, може усунути її самостійно, виконавши роботу w_k за час t_k :

$$P(W) = \{w_k(K), t_k\}. \quad (3.4)$$

Якщо проблему не вдалось вирішити, то до справи береться група базової допомоги:

$$P(W) = \{w_b(B), t_b\}. \quad (3.5)$$

Надалі, якщо потреба клієнта не задоволена, замовлення передається групі технічної підтримки для виконання робіт w_s :

$$P(W) = \{w_s (S), t_s\}. \quad (3.6)$$

Якщо група технічної підтримки не змогла справитись із проблемою, її намагаються вирішити експерти:

$$P(W) = \{w_e (E), t_e \}. \quad (3.7)$$

Якщо ж проблема, що виникла у клієнта, пов'язана із зовнішнім (відносно ІТ-компанії) розробником, то для її вирішення залучають відповідних зовнішніх спеціалістів:

$$P(W) = \{w_v (V), t_v\}. \quad (3.8)$$

Таким чином, повний цикл виконання замовлення

$$P(W) = \{w_k (K), t_k ; w_b (B), t_b ; w_s (S), t_s ; w_e (E), t_e ; w_v (V), t_v\}, \quad (3.9)$$

а тривалість становить суму всіх затрачених часів:

$$T_B = t_k + t_b + t_s + t_e + t_v \quad . \quad (3.10)$$

3.1.2 Модель «роїння»

Склад потенційних членів команди підтримки «роїння» може бути описаний у вигляді формули:

$$HR = \{F_n, S_n, R_n, E_n, D_n\}, \quad (3.11)$$

де HR - загальний склад членів команди підтримки;

F_n – група першої лінії підтримки;

S_n – група швидкого реагування;

R_n – рутинна група;

E_n – технічна та експертна підтримка;

D_n – резервна група.

Клієнт при виникненні проблеми P , звертається до першої лінії підтримки для визначення пріоритетності запиту чи замовлення:

$$P(W) = \{w_f (F), t_f\}. \quad (3.12)$$

Залежно від встановленого пріоритету (високий чи звичайний) запит опрацьовується відповідною групою підтримки. Для запитів з високим рівнем пріоритету проблему вирішують члени групи швидкого реагування з допомогою технічної та експертної підтримки:

$$P(W) = \{w_s (S, E), t_s\}. \quad (3.13)$$

Запити звичайного пріоритету виконуються членами рутинної групи також у співпраці з технічною та експертною підтримкою:

$$P(W) = \{w_r (R, E), t_r\}. \quad (3.14)$$

Якщо проблема не вирішена, то до виконання запиту залучається ще резервна команда:

$$P(W) = \{w_r (R, E, D), t_r + t_d\}. \quad (3.15)$$

Очевидно, що за рахунок попереднього визначення пріоритетності завдання значно скорочується час на його виконання:

- для завдань з високим пріоритетом

$$T_{R1} = t_f + t_s, \quad (3.16)$$

- для запитів чи завдань зі звичайним рівнем пріоритету

$$T_{R2} = t_f + t_r + t_d. \quad (3.17)$$

Тобто

$$T_{R1} \leq T_{R2} < T_B. \quad (3.18)$$

3.1.3 Комбінована модель

Склад потенційних членів комбінованої команди підтримки може бути описаний у вигляді формули:

$$HR = \{U_n, G_n, F_n, Q_n, D_n\}, \quad (3.19)$$

де HR - загальний склад членів команди підтримки;

U_n – група існуючих замовлень;

G_n – група нових замовлень;

F_n – фінансова група;

Q_n – QA-інженери;

D_n – розробники та дизайнери.

Клієнт K при виникненні проблеми P , маючи доступ до інформації, спеціально сформованої для таких випадків ІТ-компанією та відкритої для ознайомлення, може усунути її самостійно, виконавши роботу w_k за час t_k :

$$P(W) = \{w_k (K), t_k\}. \quad (3.20)$$

Якщо клієнт не знаходить потрібних йому відповідей, то визначає тип запиту: нове замовлення, існуюче замовлення, або фінансові питання. Після того, як клієнт обирає потрібну йому тему запиту, його направляють у відповідну команду підтримки:

- для розгляду проблеми, пов'язаної з існуючими замовленнями

$$P(W) = \{w_u (U), t_u\}; \quad (3.21)$$

- для здійснення нових замовлень

$$P(W) = \{w_g (G), t_g\}; \quad (3.22)$$

- для вирішення фінансових питань

$$P(W) = \{w_f (F), t_f\}. \quad (3.23)$$

Залежно від замовлення (існуюча розробка чи потреба у новій) може бути потрібна допомога QA-інженерів. Особливо це важливо на етапі впровадження чи вдосконалення існуючих замовлень:

$$P(W) = \{w_u (U, Q), t_u + t_q \}. \quad (3.24)$$

Для остаточного вирішення проблеми залучаються розробники та дизайнери:

$$P(W) = \{w_u (U, Q, D), t_u + t_q + t_d \}. \quad (3.25)$$

Таким чином, за рахунок попереднього визначення напрямку вирішення проблеми скорочується час на виконання відповідного запиту чи замовлення:

- для фінансових запитів

$$T_C = t_k + t_f ; \quad (3.26)$$

- для виявлення запитів щодо нових завдань

$$T_C = t_k + t_g ; \quad (3.27)$$

- для усунення проблем із раніше виконаними замовленнями

$$T_C = t_k + t_u + t_q + t_d . \quad (3.28)$$

Тобто

$$T_C \leq T_R \leq T_B. \quad (3.29)$$

Як видно із співвідношень, комбінована модель підтримки виконання замовлень клієнтів цілком прийнятна та має часові переваги.

Отже, не тільки члени команди підтримки здатні впливати на виконання замовлень, але й сформований в ІТ-компанії підхід до вирішення проблем клієнтів та відповідна йому модель.

3.2 Нечітка система оцінювання впливу команди підтримки на виконання замовлень

Теорія нечітких множин дає можливість застосувати для прийняття рішень щодо вибору моделі команди підтримки виконання замовлень ІТ-компанією неточні та суб'єктивні експертні знання про предметну область без формалізації їх у вигляді традиційних математичних моделей.

Для запитів чи замовлень в умовах невизначеності застосовуваної моделі команди підтримки характерними є різні терміни їх виконання. Тому в цій ситуації найбільш доцільним є застосування апарату нечіткої логіки (fuzzy logic) [43].

Перевагами fuzzy-систем у порівнянні з іншими є:

- можливість оперувати вхідними даними, заданими нечітко: наприклад, значення, що постійно змінюються в часі (динамічні задачі), або значення, що неможливо задати однозначно;
- можливість нечіткої формалізації критеріїв оцінки і порівняння: оперування критеріями «більшість», «можливе», «переважно» і т.д.;
- можливість проведення якісних оцінок як вхідних даних, так і виведених результатів, адже здійснюється оперування не тільки власне значеннями даних, але їхнім ступенем вірогідності і її розподілом;
- можливість проведення швидкого моделювання складних динамічних систем і їх порівняльний аналіз із заданим ступенем точності: оперуючи принципами поведінки системи, описаними fuzzy-методами, по-перше, не

витрачається багато часу на з'ясування точних значень змінних і складання рівнянь, що їх описують, по-друге, можна оцінити різні варіанти вихідних значень.

Загальна структура управління командою підтримки виконання замовлень, що використовує нечітку логіку, показана на рисунку 3.1.

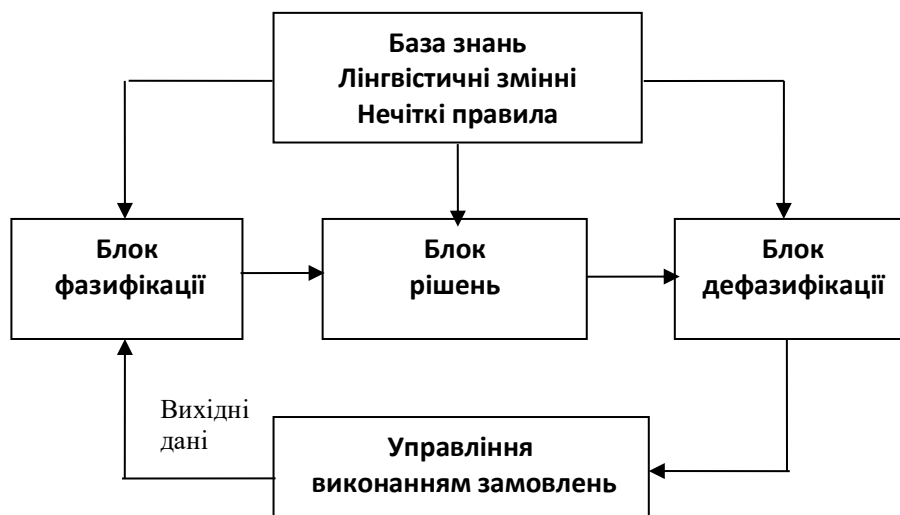


Рисунок 3.1 - Загальна структура нечіткого управління

Загальна структура нечіткого управління містить у своєму складі такі складові: блок фазифікації; база знань; блок рішень; блок дефазифікації [43].

Блок фазифікації перетворює чіткі величини, виміряні на виході об'єкта управління (в даному випадку виконанням замовлень ІТ-компанією), на нечіткі величини, описані лінгвістичними змінними у базі знань.

Блок рішень використовує нечіткі умовні (if – then) правила, закладені у базі знань, для перетворення нечітких вхідних даних на необхідні керуючі впливи, що мають також нечіткий характер.

Блок дефазифікації перетворює нечіткі дані з виходу блоку рішень на чітку величину, яка подається на виконавчий пристрій для контролю виконання замовлень [43].

Передбачається, що модель залежності вхідних та вихідних даних задана нечіткою базою знань Мамдані [43].

До переліку впливаючих на виконання запитів чи замовлень факторів рекомендується вносити важливість запиту/завдання клієнтів ІТ-компанії, тобто його пріоритет, а також існуючу в ІТ-компанії модель команди підтримки.

Нечітка система оцінювання впливу команди підтримки на виконання завдань має вхідними значеннями рівень пріоритету завдання чи запиту та модель команди підтримки.

Виходом розробленої нечіткої системи оцінювання впливу команди підтримки на виконання запитів/завдань є вчасність їх виконання.

В загальному, нечітка система оцінювання впливу команди підтримки на виконання запитів/завдань має вигляд, поданий на рисунку 3.2.



Рисунок 3.2 – Нечітка система оцінювання впливу команди підтримки на виконання запитів чи завдань

Для побудови та перевірки правильності роботи розробленої нечіткої системи використовується засіб Fuzzy Logic Toolbox середовища MATLAB. Загальна схема даної нечіткої моделі подана на рисунку 3.3.

Для вхідної змінної, що визначає пріоритет запиту/завдання, пропонується наступний розподіл:

- високий;
- звичайний;
- низький.

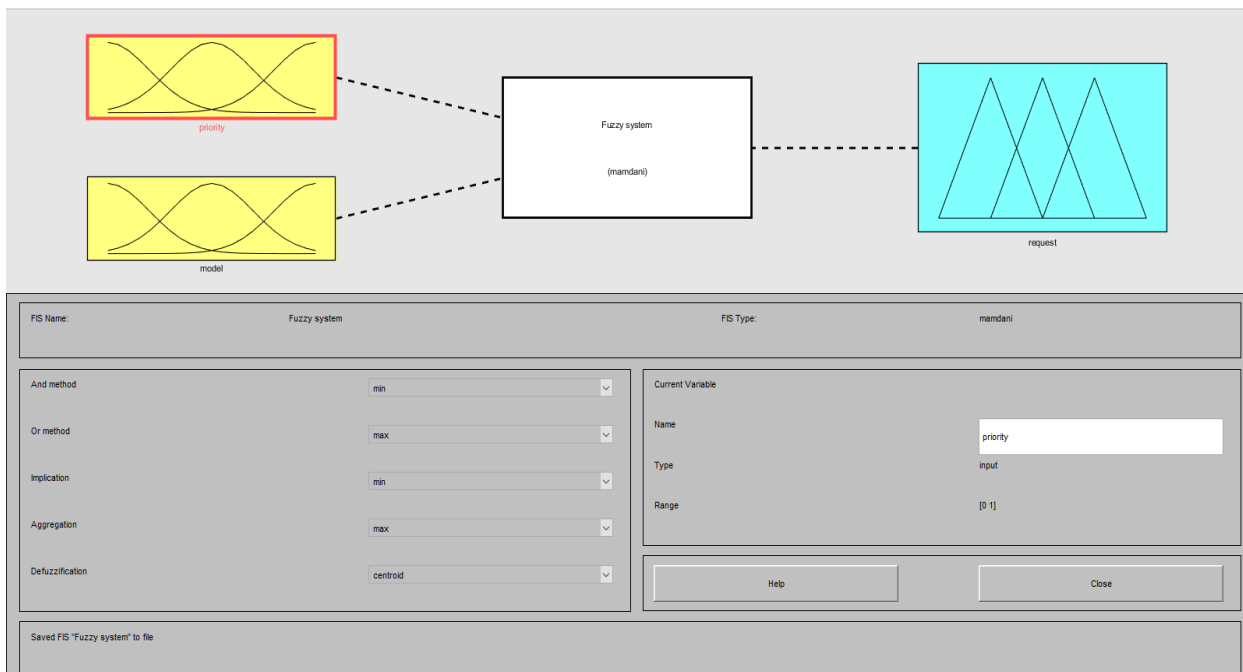


Рисунок 3.3 – Нечітка система, розроблена в середовищі MATLAB

Для вхідної змінної «модель команди підтримки» пропонується розглядати:

- багаторівнева модель;
- модель роїння;
- комбінована модель.

Для спрощення побудови нечіткої системи та з метою пришвидшення її роботи варто обрати наступний розподіл стану виконання замовлень:

- вчасне;
- з затримкою;
- швидко.

Саме такий розподіл використовується для задання вхідних змінних, тобто для побудови їх функцій належності.

Для розгляду результатів розробки та функціонування системи нечіткого висновку використовуються графічні засоби пакета Fuzzy Logic Toolbox. Ці ж засоби використовуються і при розробці систем нечіткого висновку як графічна об'єктно-орієнтована мова автоматичного програмування.

До складу цих засобів входять:

- редактор систем нечіткого висновку FIS Editor;

- редактор функцій приналежності систем нечіткого висновку Membership Function Editor (MFE);

- редактор правил систем нечіткого висновку Rule Editor;

- програма перегляду правил системи нечіткого висновку Rule Viewer;

- програма перегляду поверхні нечіткого висновку Surface Viewer.

При розробці математичних моделей на базі нечіткої логіки одним із основних та важливих етапів моделювання є вибір методу побудови функцій належності, за допомогою яких формалізуються нечіткі терми.

У випадку розроблюваної системи функції належності вхідних змінних доцільно задавати дзвоноподібною формою, що буде максимально точно їх описувати.

Нечітка система розподілу стану виконання запитів на вхід отримує значення:

- пріоритету запиту чи завдання (*priority*);

- моделі команди підтримки (*model*).

Виходом нечіткої системи є виконання запиту чи замовлення (*request*).

Для задання функцій належності кожної змінної необхідно спочатку визначити їх інтервал значень.

Для запропонованої нечіткої системи пріоритету запиту чи завдання можна задати, наприклад, в діапазоні $[0, 1]$, розділивши його на підмножини:

- високий (*high*) - $[0,5; 1]$;

- звичайний (*normal*) – $[0,2; 0,8]$;

- низький (*low*) - $[1; 0,5]$.

Задання функцій належності вхідної змінної *priority* засобами Membership Function Editor середовища MATLAB подано на рисунку 3.4.

Вхідна змінна нечіткої системи *model* буде задаватися наступними нечіткими множинами:

- багаторівнева (*multy-level*) - $[1; 0,5]$;

- роїння (*swarming*) - $[0,2; 0,8]$;

- комбінована (*combined*)- $[0,5; 1]$;

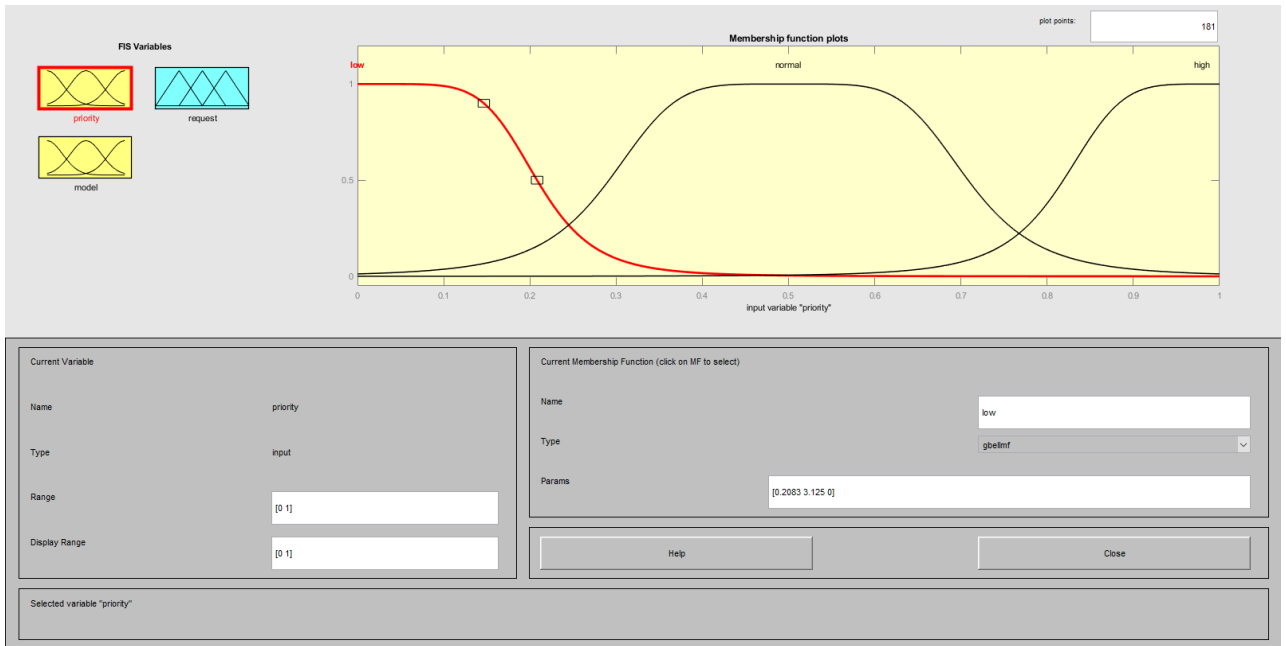


Рисунок 3.4– Функції належності вхідної змінної *priority*

Побудовані функції належності вхідної змінної *model* зображено на рисунку 3.5.

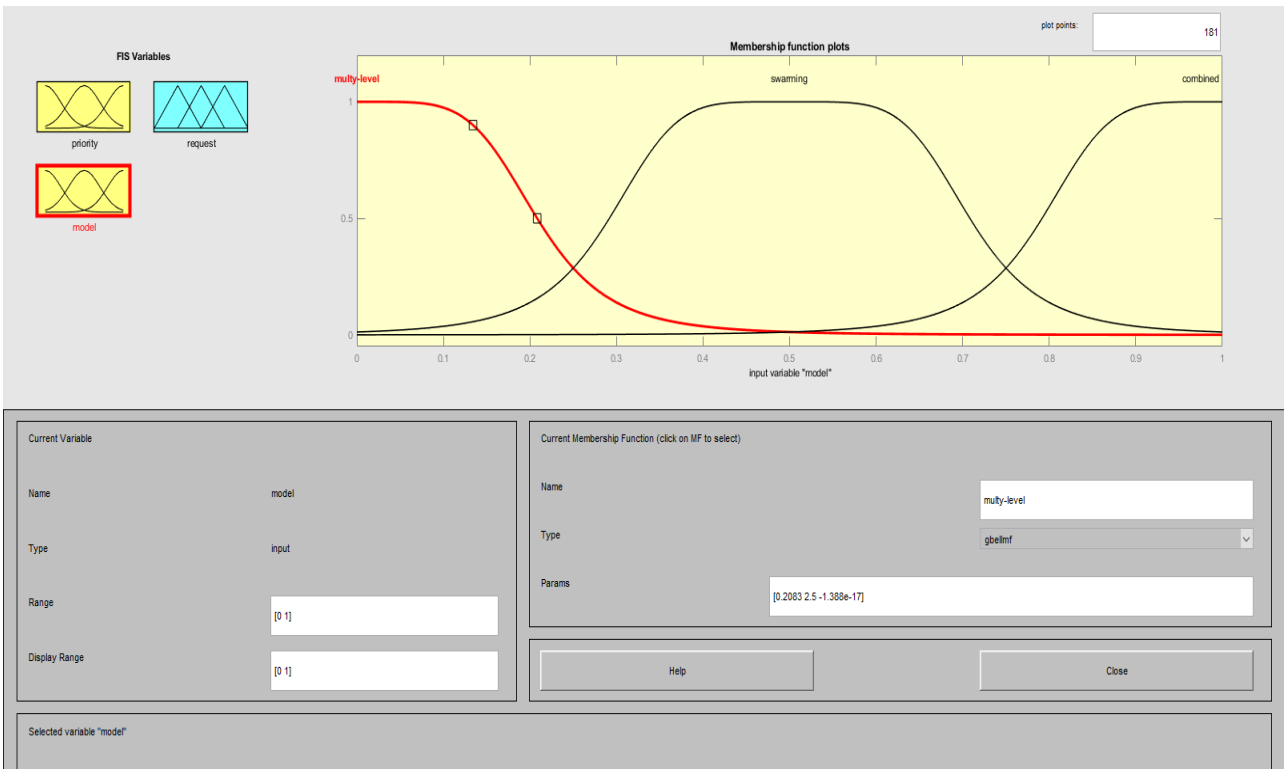


Рисунок 3.5 – Функції належності вхідної змінної *model*

Функції належності виходу розробленої нечіткої системи задаються трапецієподібною формою (рисунок 3.6), що дасть можливість точнішого проведення дефазифікації.

В даному випадку вихід нечіткої системи інтерпретує виконання запиту чи завдання. Нечітка множина *speedy*, що задається на інтервалі $[0,2; 0,8]$, вказує на швидке виконання запиту чи замовлення.

Якщо виконання запиту відбувається вчасно, то вихідна змінна нечіткої системи *request* задається множиною *timely*, що відображає інтервал значень $[0,7; 1]$. Якщо ж виконання запиту відбувається з затримкою, то вихід розробленої нечіткої системи задається множиною *with delay*, що має інтервал значень $[1; 0,3]$ (див. рисунок 3.6).

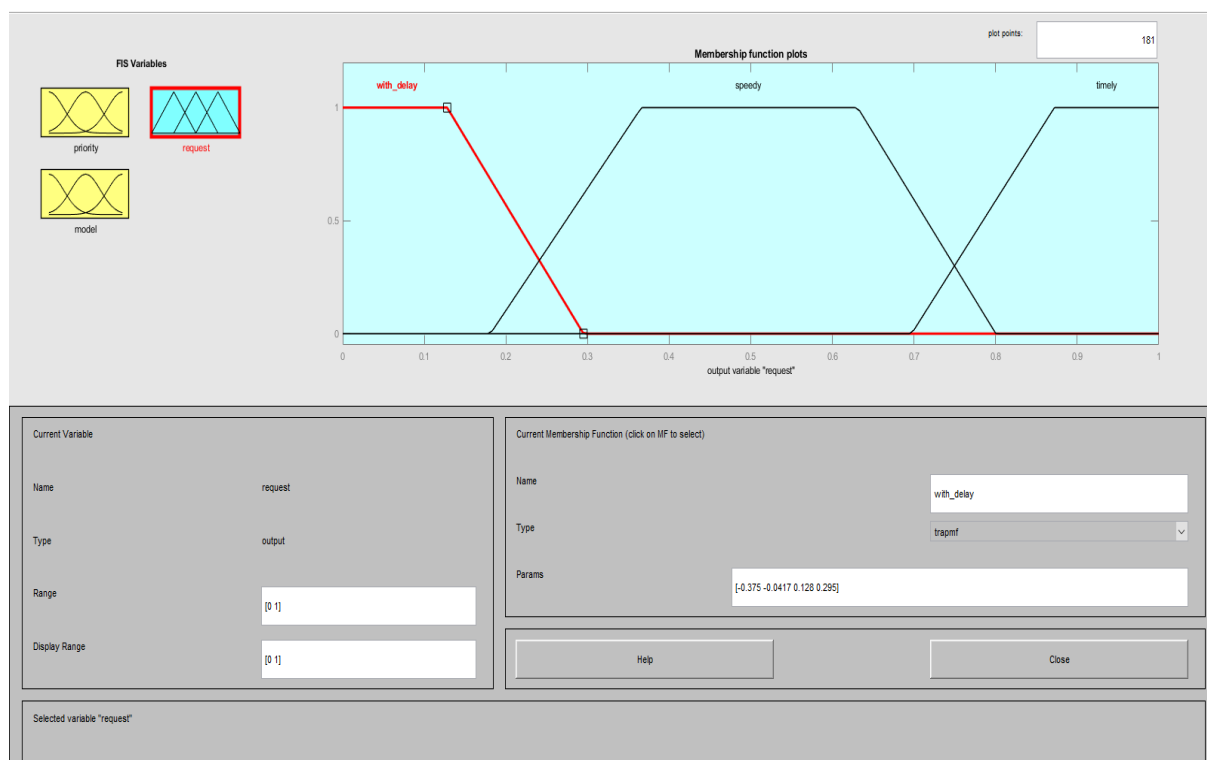


Рисунок 3.6 – Функції належності виходу *request*

Робота нечіткої системи розподілу виконання запитів/завдань залежить від бази правил. Оскільки вхідні змінні *model* та *priority* задані трьома функціями належності, і враховуючи ситуацію, коли пріоритет запиту та модель команди не задані комп'ютерною системою, то база правил складається з $R=4 \cdot 4 - 1 = 15$ правил

типу «якщо - то». Випадок, коли не задані всі вхідні змінні, не може розглядатися, оскільки тоді нечітка система не може видати висновок щодо виходу.

База правил будується відповідно до даних таблиці 3.1.

Таблиця 3.1 – Відповідність вхідних та вихідної змінних розробленої нечіткої системи

Пріоритет запити/замовлення	Модель команди підтримки	Виконання запити
Високий	Багаторівнева	З затримкою
Високий	Роїння	Швидке
Високий	Комбінована	Вчасне
Високий	Не обрана	З затримкою
Звичайний	Багаторівнева	З затримкою
Звичайний	Роїння	Швидке
Звичайний	Комбінована	Вчасне
Звичайний	Не обрана	З затримкою
Низький	Багаторівнева	Вчасне
Низький	Роїння	Швидке
Низький	Комбінована	Швидке
Низький	Не обрана	З затримкою
Не встановлений	Багаторівнева	З затримкою
Не встановлений	Роїння	Вчасне
Не встановлений	Комбінована	Вчасне

Наприклад, якщо пріоритет *запиту* приймає значення, що належить множині *високий*, модель команди багаторівнева, то можна передбачити, що виконання запити буде з затримкою.

Тобто правило, що описує даний випадок, буде мати вигляд:

If (priority is high) and (model is multy-level) then (request is with delay)

У засобі Fuzzy Logic Toolbox дана база правил має вигляд, поданий у додатку

А.

При побудові даної бази правил використовується відповідність вхідних та вихідної змінних розробленої нечіткої системи (див. таблицю 3.1).

Усі правила мають однаковий пріоритет, тому кожне з них має вагу (weight), рівну 1.

Для перевірки правильності роботи розробленої нечіткої системи виконання запитів використовується програма перегляду правил системи нечіткого висновку.

Вигляд бази правил зображено на рисунку 3.7.

Для візуалізації залежності вихідної змінної від вхідних використовується програма перегляду поверхні нечіткого висновку (рисунок 3.8).

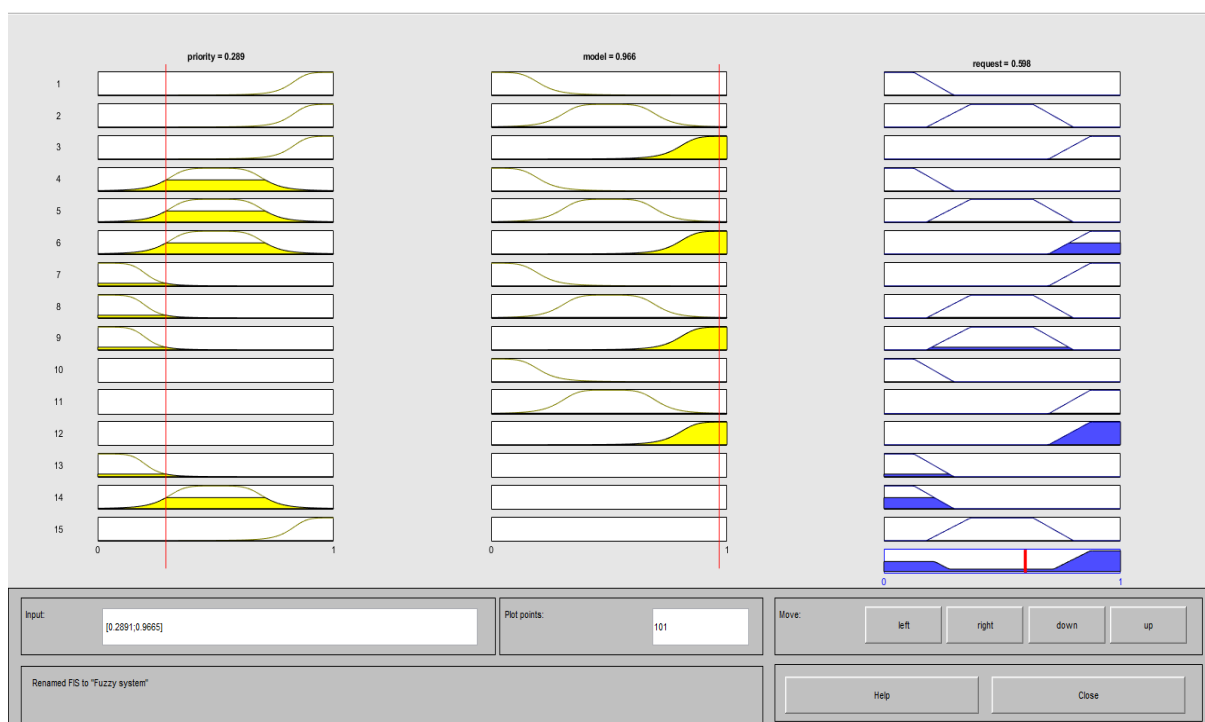


Рисунок 3.7 – Результат роботи нечіткої системи

Код розробленої нечіткої системи для визначення стану виконання запитів / замовлень представлено у додатку Б.

За допомогою даного коду можна змінювати значення функцій належності, а також самі входи і вихід запропонованої нечіткої системи, що дає змогу враховувати вплив наявної моделі команди підтримки на стан та час виконання замовлень.

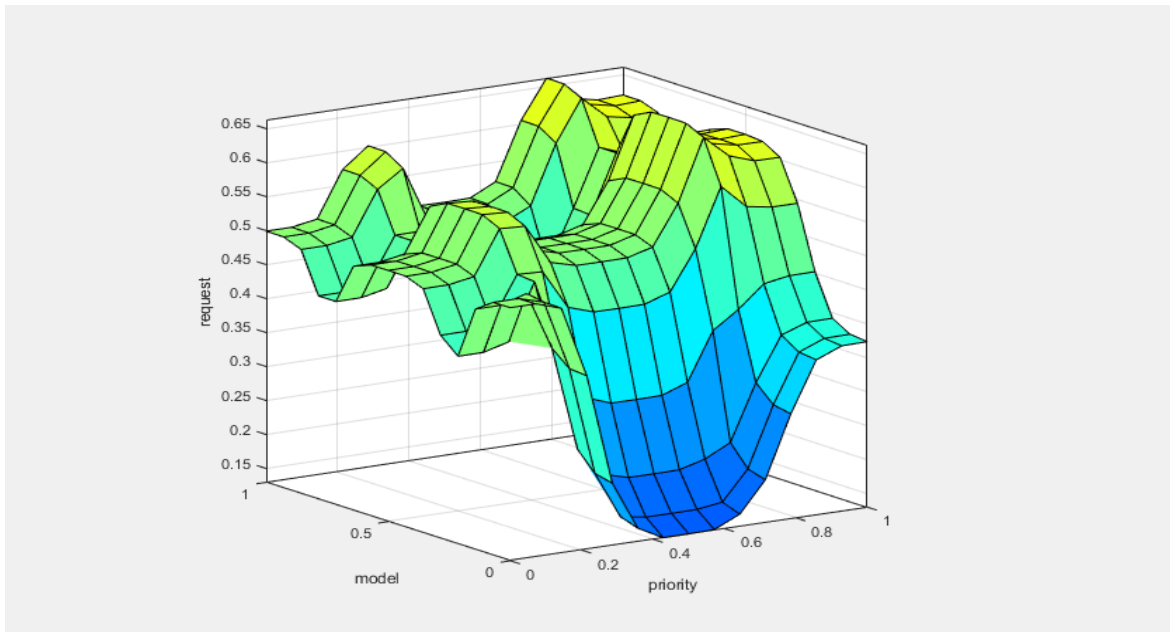


Рисунок 3.8 - Поверхня залежності виходу нечіткої системи *request* від вхідних змінних *model* та *priority*

Моделювати ситуацію можна:

- в будь-який момент виконання замовлення;
- при необхідності оцінювання стану виконання замовлень;
- при раптовій зміні складу команди підтримки виконання замовлень;
- при розгляді існуючої моделі команди підтримки виконання замовлень та аналізі перспективи її вдосконалення.

Запропонований метод дасть змогу керівництву ІТ-компанії, не володіючи достатньо точною і чіткою інформацією про можливі надходження запитів чи замовлень від клієнтів, оцінити вплив команди підтримки на їх виконання.

3.3 Практичне застосування комбінованої моделі

У висококонкурентному, цифровому світі будь-якій компанії як ніколи важливо надавати своїм клієнтам відповідну підтримку, яка швидко реагує на їх запити та вирішить будь-які проблеми.

Проаналізуємо моделі підтримки виконання замовлень на прикладі ІТ-компанії «Boosta» (м. Львів). Компанія «Boosta» займається розробкою різних ІТ-

продуктів – від сайтів та онлайн послуг до різних програм та мобільних додатків, також на вимогу клієнтів компанія може додавати новий чи змінювати старий функціонал при потребі. Крім того, надає послуги із супроводу та підтримки цих продуктів, тому відділ команди підтримки дуже великий.

Незалежно від того, чи йдеться про обмін повідомленнями через листи на електронну пошту, спілкування в онлайн-чаті чи використання соціальних мереж, або дзвінки на телефон, бути там, де клієнтам зручно просити про допомогу, а команді підтримки допомагати їм вирішити їхні проблеми — це є першочерговим пріоритетом для компанії.

Основний ресурс, який команда підтримки використовує для спілкування з клієнтами – це онлайн-чат. Впровадження онлайн-чату у процес роботи команди підтримки дає змогу допомагати клієнтам там, де їм зручніше спілкуватися. А клієнти, яким зручно користуватися послугами, як правило, купують більше і дорожче та залишаються лояльними.

Програмне забезпечення для живого чату – це технологія, яка підтримує платформи обміну миттєвими повідомленнями, за допомогою яких команда допомоги спілкується з потенційними клієнтами, клієнтами та співробітниками. Ці платформи дозволяють агентам підтримки і кінцевим користувачам вести через веб-сайт компанії, канали соціальних мереж (наприклад, Facebook Messenger), текстові повідомлення або мобільний додаток компанії.

У час, коли клієнти потребують допомоги, вони очікують від команд підтримки зручності, швидкості та персоналізованого підходу. Програмне забезпечення для живого чату та обмін повідомленнями забезпечують усі ці три функції. Але це має позитивні наслідки не лише для внутрішніх і зовнішніх клієнтів, це також є перевагою для агентів підтримки, які їх обслуговують.

Асинхронна природа обміну повідомленнями не потребує постійної присутності учасників, тобто як клієнти, так і внутрішні співробітники, які їх обслуговують — можуть робити інші речі, поки чекають відповіді один одного. Це дуже важливо для користувачів, які перебувають у інших часових поясах, коли зв'язок за допомогою телефону неможливий або незручний. Це дає змогу командам

обслуговування розширюватися та продовжувати надавати якісні послуги великій кількості людей одночасно і без збільшення чисельності персоналу.

Оскільки програмне забезпечення для живого чату розгорнуто в гнучкому, швидкому та централізованому робочому просторі, воно має такі переваги:

- масштабованість: взаємодія «один до багатьох» дозволяє агентам підтримки одночасно обслуговувати кількох клієнтів;

- швидкість: автоматизовані робочі процеси та параметри самообслуговування дозволяють користувачам вирішувати прості проблеми (наприклад, перевіряти баланс чи ціни, змінювати замовлення, перевіряти його статус, тощо);

- особисті: спілкуватися з агентами служби підтримки дуже схоже на спілкування клієнтів з родиною та друзями, бо воно відбувається у зручному для них форматі.

Компанія також використовує різні допоміжні функції живого чату, щоб, наприклад, відповідати на запити в неробочий час, або відповідати швидше, чи заздалегідь збирати ключову контактну інформацію від клієнта. Це можуть бути такі функції:

- 1) чат-боти – це комп'ютерні програми, які можуть спілкуватися з клієнтом на веб-сайті без активного агента підтримки. Чат-боти є природним доповненням до програм веб-сайтів для живого чату, оскільки вони можуть самостійно обробляти прості та повторювані запити та передавати складніші запити живим агентам, що допомагає командам масштабуватись і ефективніше обслуговувати більше клієнтів;

- 2) проактивний чат – такі чати дозволяють агентам або продавцям ініціювати чат у режимі реального часу з відвідувачем веб-сайту, що може сприяти стимулюванню та полегшенню продажів протягом усього процесу оформлення замовлення;

- 3) інтелектуальні тригери — це заздалегідь визначені правила, які визначають, що має статися, якщо виникнуть певні обставини. Наприклад, можна встановити інтелектуальний тригер, який призведе до того, що вікно чату веб-сайту

з'явиться з шаблонним повідомленням, якщо відвідувач веб-сайту проводить певний час на певній сторінці;

4) готові (шаблонні) відповіді – це попередньо написані частини тексту, призначені для використання в заздалегідь визначеній ситуації. Оскільки агенти служби підтримки часто стикаються з подібними ситуаціями, вони можуть створювати готові відповіді, щоб автоматично виконувати такі дії, як: привітати клієнта, який повернувся, запропонувати знижку, передати чат іншому агенту, дати відповідь на поширене запитання.

Для обробки клієнтських запитів компанія обрала Zendesk Chat [19]. Це чатове програмне забезпечення, яке дозволяє команді спілкуватися з клієнтами в чатах з повною історією розмов, доступною в будь-який час, включає вбудовану інтеграцію з найпопулярнішими соціальними каналами обміну повідомленнями, веб-сайтом і чатом у додатку, а також має потужні інструменти для створення чат-ботів, автоматизації та робочих процесів, його легко автоматизувати та швидко масштабувати під потреби кожного клієнта чи працівника.

Крім того, у Zendesk Chat є можливість налаштування чатового вікна відповідно до власного бренду – змінити кольори вікна, додати логотип або навіть, при необхідності, змінити логіку роботи самого чату за допомогою інструментів сторонніх розробників.

Крім Zendesk Chat, компанія також використовує Answer Bot – чат-бот Zendesk зі штучним інтелектом, який може допомогти клієнтам із самообслуговуванням, надаючи прямі посилання на статті довідкового центру або передаючи розмову агенту разом із усім контекстом та інформацією, якою клієнт уже поділився, що значно пришвидшує роботу працівника служби підтримки.

Зрештою, для ефективного результату роботи команди підтримки менеджерам потрібно постійно слідкувати за продуктивністю агентів, кількісними характеристиками їх роботи (наприклад, кількість прийнятих чатів, швидкість відповіді на повідомлення від клієнта або середня тривалість чату) та статистичними даними. Для цього компанія використовує попередньо створену та вбудовану в чатову систему інформаційну панель аналітики.

За допомогою цієї інформаційної панелі можна отримати інформацію про:

- усі сеанси чату, створені агентом та отримані від клієнта – є можливість відфільтрувати кожен звіт за датою, відділом, типом, швидкістю обробки запиту та іменем агента, який розпочав чат;

- ефективність агентів під час спілкування з клієнтом за допомогою чату – є доступ до таких показників, як: середні показники часу очікування, часу відповіді та кількості повідомлень у чаті;

- задоволеність клієнтів після сеансу чату – чати з хорошою/поганою оцінкою, позитивні та негативні відгуки клієнтів про роботу команди підтримки;

- загальну взаємодію агентів підтримки з клієнтами – загальна кількість чатів, середня тривалість кожного чату, середня кількість повідомлень, надісланих агентом під час чату.

Для будь-якого бізнесу клієнтські дані дуже важливі – але лише тоді, коли можливо їх упорядкувати та використовувати. Саме тут на допомогу приходить програмне забезпечення для керування взаємовідносинами з клієнтами. Незалежно від того, чи хочете потрібно повідомити клієнтів про нову спеціальну пропозицію щодо одного з продуктів, чи є мета забезпечити певну кількість нових клієнтів протягом місяця – усе це починається з організованої системи.

Управління взаємовідносинами з клієнтами (CRM) — це бізнес-стратегія, яка використовується компаніями, які хочуть використовувати технології для кращого управління своїми відносинами з потенційними клієнтами та вже існуючими клієнтами. Дана компанія використовує самостійно створену програму як частину своєї стратегії CRM.

Програмне забезпечення CRM – це платформа, яка допомагає компанії керувати відносинами з клієнтами, об'єднуючи команди та організовуючи нотатки, завдання та показники в одному місці. Ця платформа діє як додаток для керування контактами, тобто у ній зберігається вся інформація про клієнтів, як потенційних, так і вже існуючих, наприклад:

- номери телефонів;
- електронні адреси;

- контакти у соціальних мережах;
- адреси;
- історія замовлень та оновлень;
- історія відгуків;
- програми лояльності.

Навіть якщо потенційний клієнт не приносить продажі, програмне забезпечення фіксує кожен взаємодію клієнта з веб-сайтом через низку каналів. Оскільки CRM-система та Zendesk Chat синхронізовані між собою, це звільняє представників команди підтримки від ручного введення даних і дає їм багато корисного ресурсу, який можна використовувати для персоналізації майбутніх повідомлень та контакту з цими потенційними клієнтами.

Переваги використання CRM-системи у роботі команди IT-підтримки:

- 1) централізована платформа містить всю інформацію про усіх клієнтів в одному місці, щоб агенти команди підтримки завжди мали повний актуальний контекст клієнта;
- 2) точне прогнозування обсягу угод та зростання доходу, щоб була можливість вдосконалювати стратегію розвитку компанії;
- 3) аналіз процесу та прогресу угод, що дає змогу передбачити, де можуть бути слабкі місця, які потребують покращення;
- 4) CRM-система збирає, впорядковує та зберігає усі дані клієнтів, щоб була можливість створювати цільові персоналізовані кампанії для аудиторії, сегментувати клієнтів, щоб краще розуміти, як адаптувати досвід для кожної групи;
- 5) різні команди можуть легко співпрацювати, коли кожен використовує актуальні дані з інформаційної панелі продажів. Клієнти не люблять повторюватися, тому коли відділи обмінюються інформацією всередині себе, клієнтський досвід значно покращується;
- 6) використання CRM-системи для автоматизації повторюваних завдань допомагає бізнесу забезпечити чудовий досвід клієнтів, звільнивши команду підтримки та зосередивши їх на продажах.

На ринку послуг є величезна кількість різноманітних CRM-програм, які

можна інтегрувати у свій бізнес. Але компанія створила свою власну унікальну CRM-систему, що дає змогу змінювати її відповідно до потреб бізнесу, робити оновлення та покращення не залучаючи зовнішніх ресурсів та не витрачаючи зайвого часу. Розробкою та вдосконаленням CRM-системи займається окрема група розробників.

Наявність кількох каналів комунікації з клієнтами для залучення та взаємодії є потребою сучасності для підвищення рівня клієнтського досвіду. Тому другим каналом зв'язку з клієнтами у компанії є дзвінок. Насправді, телефонні дзвінки завжди були чудовим каналом комунікації з клієнтами для обговорення важливих чи термінових питань та надання швидкої відповіді. Телефонний дзвінок можна використовувати для ділового спілкування у випадках, коли немає необхідності ділитися або обмінюватися візуальними матеріалами з клієнтами.

Недоліком традиційного телефонного дзвінка є те, що один агент підтримки може обслуговувати лише один дзвінок, а клієнтам часто доводиться чекати в довгих чергах, перш ніж вони зможуть поговорити з консультантом і це є надзвичайно виснажливим і незручним. Тим не менше, більшість клієнтів добре знайомі з цією концепцією комунікації, тому телефонний дзвінок досі залишається одним із найбільш звичних варіантів для команд обслуговування клієнтів.

Електронна пошта є третім каналом комунікації між клієнтом на команду IT-підтримки. Це економічно ефективний спосіб охопити цільову аудиторію та підтримувати постійні відносини.

Це універсальний канал зв'язку, який можна використовувати для маркетингу, обміну новинами, особистого листування, вирішення клієнтських проблем тощо у тих випадках, коли негайна відповідь не потрібна.

Один із найефективніших способів надання допомоги електронною поштою – це відповідати на електронні листи між взаємодіями в реальному часі на інших контактних каналах. Наприклад, якщо консультант перебуває між двома розмовами в чаті, він може тим часом відповідати на запити електронною поштою.

Оскільки електронна пошта синхронізована із CRM-системою, це дає кращу видимість процесу продажів і підвищує продуктивність за допомогою сповіщень,

шаблонів, автоматизації тощо. Таким чином, можна створювати цілі промо-кампанії через електронну пошту – автоматично надсилати електронні листи потенційним клієнтам або певним групам клієнтів через вибрані інтервали часу.

З точки зору недоліків, одна з головних скарг клієнтів щодо електронної пошти полягає в тому, що вона має низький середній час відповіді.

У компанії команда IT-підтримки – це великий компонент, але це не єдиний тип команди обслуговування клієнтів в організації. Організаційна структура команди обслуговування клієнтів – це структура ролей, команд і рівнів, які допомагають розбудувати підхід до обслуговування клієнтів, орієнтований в першу чергу на задоволеність клієнта.

У компанії команда підтримки має таку організаційну структуру:

1) спеціалісти команди підтримки. Ця команда є найбільшою у всій структурі і працює вона лише з клієнтами, які уже мають замовлення у компанії. Спеціалісти служби підтримки відповідають на телефонні дзвінки, електронні листи, повідомлення в чаті та на повідомлення в соціальних мережах;

2) спеціалісти команди продажів. Ця команда схожа на команду підтримки, але вона працює лише з потенційними клієнтами. Спеціалісти з продажів відповідає на запитання нових користувачів, допомагає зі створенням замовлення, консультує щодо вибору потрібного сервісу чи продукту;

3) спеціалісти з фінансової підтримки клієнтів, які допомагають з проблемами під час оплати замовлень, повернення коштів тощо;

4) експерти з контролю якості продуктів. Ця команда спеціалістів гарантує, що продукти чи послуги, які отримують клієнти, завжди відповідають або перевершують очікування, а це у свою чергу сприяє підвищенню репутації компанії. Якщо представник служби підтримки клієнтів не має достатньо глибоких знань про те, вирішити ту чи іншу проблему, команда експертів із контролю якості знайде рішення та відповідь на ці запитання;

5) команда підтримки розробників. Ця команда працює з розробкою, керуванням оновленнями, виправленням помилок всередині коду тощо.

Для успішної роботи та якісної взаємодії між різними командами та

структурами компанія користується послугами Slack [20]. Slack – це програма обміну повідомленнями для бізнесу, яка пов’язує людей з необхідною інформацією. Slack організовує розмови у виділених просторах, які називаються каналами. Канали вносять порядок і ясність у роботу — можна створити їх для будь-якого проекту, теми чи команди. Маючи потрібних людей і інформацію в одному місці, команди можуть обмінюватися ідеями, приймати рішення та просувати роботу.

Виняткове обслуговування клієнтів є потужним бізнес-активом. Компанії, які постійно надають відмінне обслуговування клієнтів, отримують переваги від вищих доходів, утримання клієнтів, зіркової репутації, кращої стійкості, ніж у конкурентів, і глибокого розуміння продукту.

Висновки до розділу 3

1. Проведені дослідження багаторівневої моделі, моделі роїння та комбінованої моделі IT-підтримки на основі їх математичного опису вказують на те, що комбінована модель підтримки виконання замовлень клієнтів цілком прийнятна та має часові переваги.

2. Побудовано нечітку систему оцінювання впливу команди підтримки на виконання замовлень, яка на вхід отримує значення рівня пріоритету запиту і модель команди підтримки. Виходом нечіткої системи є стан виконання запиту/завдання. Для побудови та перевірки правильності роботи розробленої нечіткої системи використано засіб Fuzzy Logic Toolbox середовища MATLAB.

3. Розроблено базу правил нечіткої системи та проведено дослідження нечіткої системи оцінювання впливу команди підтримки на виконання замовлень. Запропонований підхід дасть змогу керівництву IT-компанії, не володіючи достатньо точною і чіткою інформацією про можливі надходження запитів чи замовлень від клієнтів, оцінити вплив команди підтримки на їх виконання.

4. Описано переваги практичного застосування комбінованої моделі управління командою підтримки виконання замовлень IT-компанії.

ВИСНОВКИ

1. Команди підтримки клієнтів ІТ-компаній відіграють чи не одну з найважливіших ролей у процесі замовлення та супроводу будь-якого ІТ-продукту чи послуги. Для взаємодії служби підтримки з клієнтами потрібно використовувати різні моделі управління, які повинні допомагати ефективно керувати наявними людськими ресурсами у службі підтримки для швидкого та якісного виконання запитів клієнтів.

2. Проаналізовано сучасні підходи до управління командами підтримки у ІТ-компаніях. Дослідження показало, що впродовж кількох останніх десятиліть, найпопулярнішою моделлю є багаторівневий метод управління командою підтримки, але в сучасних умовах виникла потреба у створенні більш адаптивних, ефективних та гнучких методів для оптимізації процесу підтримки, скорочення часу вирішення проблем та збереження фінансових ресурсів компаній.

3. Проведено дослідження багаторівневої моделі управління командами ІТ-підтримки та моделі «роїння».

4. Розроблено й описано схеми та алгоритми діяльності команди підтримки відповідно до цих моделей. Розглянуто переваги та недоліки кожної з моделей, приведені приклади труднощів, які можуть виникнути при їх імплементації.

5. На основі проведеного аналізу багаторівневої та моделі роїння запропоновано комбіновану модель управління ІТ-підтримкою, в якій за основу взято архітектуру моделі «роїння» з елементами багаторівневої моделі. Це дозволить використовувати переваги обох моделей та ефективніше вирішувати завдання, поставлені перед відділом ІТ-підтримки компанії.

6. Розроблено схему та алгоритм комбінованої моделі підтримки.

7. Проведені дослідження багаторівневої моделі, моделі «роїння» та комбінованої моделі ІТ-підтримки на основі їх математичного опису вказують на те, що комбінована модель підтримки виконання замовлень клієнтів цілком прийнятна та має часові переваги.

8. Побудовано нечітку систему оцінювання впливу команди підтримки на виконання замовлень, яка на вхід отримує значення рівня пріоритету запиту і модель команди підтримки. Виходом нечіткої системи є стан виконання запиту/завдання. Для побудови та перевірки правильності роботи розробленої нечіткої системи використано засіб Fuzzy Logic Toolbox середовища MATLAB.

9. Розроблено базу правил нечіткої системи та проведено дослідження нечіткої системи оцінювання впливу команди підтримки на виконання замовлень. Запропонований підхід дасть змогу керівництву ІТ-компанії, не володіючи достатньо точною і чіткою інформацією про можливі надходження запитів чи замовлень від клієнтів, оцінити вплив команди підтримки на їх виконання.

10. Описано переваги практичного застосування комбінованої моделі управління командою підтримки виконання замовлень ІТ-компанії.

11. Розроблена комбінована модель може бути використана для ефективного управління командами ІТ-підтримки при обслуговуванні клієнтів та вирішенні вхідних запитів (додаток В).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. What is customer care? Definition, importance, and best practices [Електронний ресурс] / P. Alig // 2021– Режим доступу: <https://www.zendesk.de/blog/guide-putting-care-back-customer-care/#georedirect>.
2. What is a custom software development company and how to choose one [Електронний ресурс] – Режим доступу: <https://dac.digital/what-is-a-custom-software-development-company-and-how-to-choose-one/>.
3. Everything You Should Know Before Hiring a Software Development Company [Електронний ресурс] / M. Warcholinski – Режим доступу: <https://brainhub.eu/library/before-hiring-software-company>.
4. Сметанюк О. А. Особливості системи управління проектами в іт-компаніях / О.А. Сметанюк, А.В. Бондарчук – 2020. – № 10. – С. 105-111.
5. Малєєва Ю. А. Інформаційне та програмне забезпечення менеджера з персоналу ІТ-компанії / Ю.А. Малєєва, О.Ю. Персіянова, В.В. Косенко – 2018. – С. 22-32.
6. Замула І. В. ІТ-послуга: поняття та види для облікових цілей / І.В. Замула, Л.В. Чижевська, І.Л. Грабчук // Проблеми теорії та методології бухгалтерського обліку, контролю і аналізу – 2021. – С.29-33.
7. What is ITSM? The beginner's guide to ITSM (IT service management) [Електронний ресурс] – Режим доступу: <https://www.manageengine.com/products/service-desk/itsm/what-is-itsm.html#bestpractice>.
8. A Multi-tiered Approach to Enterprise Support Services [Електронний ресурс]: Design, User Experience, and Usability. Theory, Methods, Tools and Practice: International Conference DUXU 2011 / C.S. Chandrasekaran, W.R. Simpson // Lecture Notes in Computer Science, vol 6769. – 2011 – С. 388–397. – Режим доступу: https://link.springer.com/chapter/10.1007/978-3-642-21675-6_45
9. Why Customer Service Frustrates Consumers: Using a Tiered Organizational Structure to Exploit Hassle Costs [Електронний ресурс] / A. Dukes, Y. Zhu // 2019 –

Marketing Science, – vol.38 – №3 – С. 500-515 – Режим доступу:
<https://pubsonline.informs.org/doi/10.1287/mksc.2019.1149>

10. Yıldırım N. Organisational learning through knowledge management systems: a case study on improvement of customer support processes / N. Yıldırım // International Journal of Knowledge Management Studies. – 2017 –8:3-4. – С. 375-402

11. Application of Lean Six Sigma in IT support services – a case study [Електронний ресурс] / E.V Gijo, A. Jiju, V.M. Sunder // 2019 – The TQM Journal, – vol. 31 – No. 3, – С. 417-435 – Режим доступу:
<https://www.emerald.com/insight/content/doi/10.1108/TQM-11-2018-0168/full/html>

12. Modern software product support processes and the usage of multimedia formats [Електронний ресурс] / K.C. Parmit, T. Grossman, G. Fitzmaurice // 2011 – Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'11) – С. 3093–3102 – Режим доступу:
<https://dl.acm.org/doi/10.1145/1978942.1979400>

13. Multi-domain case-based module for customer support [Електронний ресурс] / S.Heras, J.A. García-Pardo, R. Ramos-Garijo, A. Palomares, V. Botti, M. Rebollo, V. Julián // 2009 – Expert Systems with Applications – С. 6866 – 6873 – Режим доступу :
<https://doi.org/10.1016/j.eswa.2008.08.003>

14. Кухар М. А. Концептуальні моделі підтримки прийняття рішень в багаторівневих системах адміністрування / М.А. Кухар // Вісник ВПІ – 2020. – №3. – С. 51–56.

15. Call centre management: responsibilities and performance [Електронний ресурс] / G. Robinson, C. Morley // 2006 – International Journal of Service Industry Management – vol. 17 – No. 3 – С. 284-300 – Режим доступу:
<https://doi.org/10.1108/09564230610667122>

16. IT Support Levels Clearly Explained: L1, L2, L3 & More [Електронний ресурс] / C. Kidd, J. Hertvik // 2019 – Service Management Blog – Режим доступу :
<https://www.bmc.com/blogs/support-levels-level-1-level-2-level-3/>

17. Technical Support Engineer Role & Responsibilities [Електронний ресурс] / C. Kidd // 2019 – The Business of IT Blog – Режим доступу : <https://www.bmc.com/blogs/technical-support-engineer/>

18. Evaluating Technical Support Models: Tiered Support vs. Swarming, Part 1 [Електронний ресурс] / P. Dooley // 2017 – Режим доступу : <https://www.thinkhdi.com/library/supportworld/2017/evaluating-technical-support-models-tiered-support-vs-swarming-part-1.aspx>

19. Tier Support: How to Organize your Customer Service Team [Електронний ресурс] / S. Chambers // 2021 – Режим доступу : <https://freshdesk.com/customer-support-software/tier-support-blog/>

20. Шашкова Н. І. Управління проектами в ІТ сфері: застосування гнучких методологій / Н. І. Шашкова, І. Г. Фадєєва, Т. С. Казакова // Scientific notes of Lviv University of Business and Law. – 2021. – С. 166-172.

21. Борисова, Н. В. Технологія підбору команди ІТ-фахівців для виконання проекту / Н. В. Борисова, К. В. Мельник, І. В. Оліфенко // Інформаційні технології в освіті, науці і техніці : міжнар. наук.-практ. конф., 21-23 травня 2020 р. : тези доповіді – V., 2020. – С. 40.

22. Swarming vs Tiered Support Models Explained [Електронний ресурс] / J.Stevens-Hall // 2018 – Service Management Blog – Режим доступу : <https://www.bmc.com/blogs/swarming-support-tiered-support-differences/>

23. Evaluating Technical Support Models: Tiered Support vs. Swarming, Part 2 [Електронний ресурс] / P. Dooley // 2017 – Режим доступу : <https://www.thinkhdi.com/library/supportworld/2017/evaluating-technical-support-models-tiered-support-vs-swarming-part-2.aspx>

24. A nice example of Swarming in support of a Service Desk [Електронний ресурс] / J. Stevens-Hall // 2018 – Режим доступу : <https://jonstevenshall.medium.com/a-nice-example-of-swarming-in-support-of-a-service-desk-7bdbe8e9890c>

25. Are there drawbacks to Swarming? [Електронний ресурс] / J. Stevens-Hall // 2019 – Режим доступу : <https://medium.com/itrevolution/are-there-drawbacks-to-swarming-b42309e7541c>
26. Доценко Н.В. Методи управління людськими ресурсами при формуванні команд мультипроектів та програм: монографія / Н. В. Доценко, Л. Ю. Сабадош, І. В. Чумаченко // – Харків, –2015. – С. 201.
27. Zendesk [Електронний ресурс] – Режим доступу: <https://www.zendesk.com/>
28. Customer relationship management and its impact on entrepreneurial marketing: a literature review [Електронний ресурс] / V. Guerola-Navarro, H. Gil-Gomez, R. Oltra-Badenes // 2022 – Режим доступу : <https://doi.org/10.1007/s11365-022-00800-x>
29. Slack [Електронний ресурс] – Режим доступу: <https://slack.com/>
30. Bernard M. Key Performance Indicators (KPI): The 75 measures every manager needs to know / M. Bernard, 2012.
31. Рудінська О., Белякова В. Критерії й оцінка ефективної діяльності управлінської команди в сучасному багатoproфільному підприємстві / О. Рудінська, В. Белякова // Ринкова економіка: сучасна теорія і практика управління. – 2016 – 15:3(34).
32. Ганза І. В. Специфіка формування проектної команди: управлінський аспект. / І. В. Ганза // Ефективна економіка. – 2015. –№7.
33. Євтушенко Г. І. Формування команди проекту та організація її ефективної роботи (теоретичний аспект) / Г. І. Євтушенко // Східна Європа: економіка, бізнес та управління. – 2019. – №4 (21). – С.77–82.
34. Оленіч А. В. Формування і розвиток проектної команди в сучасних умовах / А. В. Оленіч, З. Я. Шацька // Економіка та управління підприємствами. – 2012. – №10. – С.136–142.
35. Марцінковська О. Сучасні підходи до управління командами в контексті реалізації проекту / О. Марцінковська // Регіональні аспекти розвитку продуктивних сил України. – 2016. – № 21. – С.102–108.

36. Продіус О. І. Формування та управління командою проекту / О. І. Продіус // Вісник ОНУ ім. І. І. Мечникова. – 2016. – №10 (52). – С.51–55.

37. Кононенко І. Формування команди проекту з розробки інформаційно-комунікаційних технологій / І. Кононенко, Г. Сушко // Інформаційні технології і засоби навчання. – 2019. – №5.

38. Москаленко В. О. Сучасні підходи до формування команди проекту / В. О. Москаленко // Бізнес та інтелектуальний капітал. – 2014. – №5. – С.78–86.

39. Орлова О. М. Особливості управління персоналом в ІТ-сфері. / О. М. Орлова // Науковий вісник Ужгородського національного університету. – 2017. – С. 117-120.

40. Лазоренко Т. В. Оцінка ефективності навчання персоналу в ІТ компаніях / Т. В. Лазоренко, Ю. К. Німуха // Молодий вчений – 2018. – №1. – С. 914-916.

41. Жавела К. А. Сучасні концепції та інноваційні технології в системі управління персоналом / К. А. Жавела, А. К. Жавела // Інвестиції: практика та досвід. – 2019. – С. 73-78.

42. Volkvadze N. Сучасні методи мотивації персоналу ІТ-компаній / N. Volkvadze, T. Hetman // Economic journal of Lesia Ukrainka Eastern European National University. – 2020. С. 72-82.

43. Fuzzy Logic Toolbox [Електронний ресурс] – Режим доступу: <https://docs.exponenta.ru/fuzzy/index.html>

44. Васильків Н.М. Модель управління командою підтримки виконання замовлень ІТ-компанією / Н. М. Васильків, А. С. Волкова // Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення: міжнар. наук.-техн. конф, м. Тернопіль, 6-7 квітня 2022 р.: збірник тез доповідей: випуск 66, 2022 – С. 9-10.

45. Васильків Н. М. Модель забезпечення якості ІТ-продукту / Н. М. Васильків, Д. В. Гаврилюк, А. С. Волкова // Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення: міжнар. наук.-техн. конф, м. Тернопіль, Україна – м. Переворськ, Польща, 15-16 листопада 2022 р.: збірник тез доповідей: випуск 72, 2022 – С.25-26.

46. Загальні рекомендації з підготовки, оформлення, захисту та оцінювання випускних кваліфікаційних робіт здобувачів вищої освіти першого «бакалаврського» і другого «магістерського» рівнів / За ред. доц. М.І.Шинкарика. Тернопіль: ТНЕУ, 2018. 67 с.

47. Комар М.П., Саченко А.О., Васильків Н.М. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за другим (магістерським) рівнем вищої освіти. Тернопіль: ЗУНУ, 2021. 32 с.

Додаток А
База правил нечіткої системи

1. If (priority is high) and (model is multy-level) then (request is with_delay) (1)
2. If (priority is high) and (model is swarming) then (request is speedy) (1)
3. If (priority is high) and (model is combined) then (request is timely) (1)
4. If (priority is normal) and (model is multy-level) then (request is with_delay) (1)
5. If (priority is normal) and (model is swarming) then (request is speedy) (1)
6. If (priority is normal) and (model is combined) then (request is timely) (1)
7. If (priority is low) and (model is multy-level) then (request is timely) (1)
8. If (priority is low) and (model is swarming) then (request is speedy) (1)
9. If (priority is low) and (model is combined) then (request is speedy) (1)
10. If (model is multy-level) then (request is with_delay) (1)
11. If (model is swarming) then (request is timely) (1)
12. If (model is combined) then (request is timely) (1)
13. If (priority is low) then (request is with_delay) (1)
14. If (priority is normal) then (request is with_delay) (1)
15. If (priority is high) then (request is speedy) (1)

Додаток Б

Текст коду

```

[System]
Name='Fuzzy system'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=15
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='priority'
Range=[0 1]
NumMFs=3
MF1='low': 'gbellmf', [0.2083 3.125 0]
MF2='normal': 'gbellmf', [0.2083 2.5 0.5]
MF3='high': 'gbellmf', [0.181216931216931 2.5 1]

[Input2]
Name='model'
Range=[0 1]
NumMFs=3
MF1='multy-level': 'gbellmf', [0.2083 2.5 -1.388e-17]
MF2='swarming': 'gbellmf', [0.2083 2.5 0.5]
MF3='combined': 'gbellmf', [0.2083 2.5 1]

[Output1]
Name='request'
Range=[0 1]
NumMFs=3
MF1='with_delay': 'trapmf', [-0.375 -0.0417 0.128 0.294973544973545]
MF2='speedy': 'trapmf', [0.181 0.366 0.631 0.80026455026455]
MF3='timely': 'trapmf', [0.697089947089947 0.872 1.04 1.38]

[Rules]
3 1, 1 (1) : 1
3 2, 2 (1) : 1
3 3, 3 (1) : 1
2 1, 1 (1) : 1
2 2, 2 (1) : 1
2 3, 3 (1) : 1
1 1, 3 (1) : 1
1 2, 2 (1) : 1
1 3, 2 (1) : 1
0 1, 1 (1) : 1
0 2, 3 (1) : 1
0 3, 3 (1) : 1
1 0, 1 (1) : 1
2 0, 1 (1) : 1
3 0, 2 (1) : 1

```

Додаток В

Додаток Г

Копії опублікованих матеріалів дослідження