

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Таврійський національний університет
імені В.І. Вернадського
Навчально-науковий інститут муніципального управління
та міського господарства
Кафедра інженерних систем та технологій

На правах рукопису

Кваліфікаційна робота на здобуття ступеня вищої освіти «бакалавр»

АВТОМАТИЗАЦІЯ ПРОЦЕСУ ОЦІНЮВАННЯ ТА ПРОГНОЗУВАННЯ
ПОКАЗНИКІВ НАДІЙНОСТІ ПРОГРАМНИХ СИСТЕМ

Здобувача вищої освіти
Клопота Ярослава Станіславовича
Освітня програма
«Автоматизація та комп'ютерно-
інтегровані технології»
(Спеціальність
174 «Автоматизація, комп'ютерно-
інтегровані технології та
робототехніка»)

(підпис)

Науковий керівник:
ст.викл. Мінаєва Ю.Ю.

(підпис)

Національна шкала _____
Кількість балів _____
Оцінка: ECTS

Таврійський національний університет імені В. І. Вернадського
Навчально-науковий інститут муніципального управління
та міського господарства
Кафедра Інженерних систем та технологій
Освітньо-кваліфікаційний рівень бакалавр
Спеціальність 174 «Автоматизація, комп'ютерно-інтегровані технології та
робототехніка»

«ЗАТВЕРДЖУЮ»
Завідувач кафедри
_____ Наталія ОМЕЦІНСЬКА
« ____ » _____ 2025 року

ЗАВДАННЯ НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ

Клопота Ярослава Станіславовича

1. Тема роботи: Автоматизація процесу оцінювання та прогнозування показників надійності програмних систем
керівник роботи: ст.викл. Мінаєва Ю.Ю.

затверджені Наказом ТНУ імені В.І Вернадського:
від «7» листопада 2025 № 138/1-ОД

2. Строк подання студентом роботи: 25.11.2025

3. Вихідні дані до роботи: масиви даних результатів тестування, загальна кількість виділених модулів, сумарна кількість змінних у кодї та показники зв'язності модулів, необхідні для вибору стратегії тестування.

4. Зміст розрахунково-пояснювальної записки: Дослідити теоретичні засади технічної діагностики, алгоритми діагностування та кількісні показники надійності сучасних інформаційно-комунікаційних систем; Запропонувати комплекс рішень для автоматизації бізнес-процесів малого підприємства (на прикладі фірми Sigma), що включає розробку програми для SPACE-аналізу, веб-сервісу діагностики та реляційної бази даних; Проаналізувати потенційні небезпеки, правила охорони праці при роботі з комп'ютерною технікою та заходи безпеки у надзвичайних ситуаціях.

5. Перелік графічного матеріалу: графічна робота виконана у вигляді мультимедійної презентації

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ 1	старший викладач Юлія МІНАЄВА		
Розділ 2	старший викладач Юлія МІНАЄВА		
Розділ 3	старший викладач Юлія МІНАЄВА		
Розділ 4	старший викладач Юлія МІНАЄВА		

7. Дата видачі завдання _____ 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Термін виконання етапів роботи	Заключний документ етапу
1	Вибір теми бакалаврської роботи, призначення керівника	до 31.10.2025	Затвердження переліку тем бакалаврських робіт
2	Пошук і відбір літератури по темі роботи, складання плану бакалаврської роботи	до 14.12.2025	Список літературних джерел, план роботи
3	Визначення об'єкта, предмета, мети, завдань та методів дослідження, написання вступу до теми бакалаврського дослідження	до 25.01.2026	Текст вступу
4	Написання тексту бакалаврської роботи відповідно до її структури:	26.01.2026-17.05.2026	Текст розділів
	4.1 I розділ	26.01.2026-6.03.2026	Текст розділу
	4.2 II розділ	9.03.2026-27.03.2026	Текст розділу
	4.3 III розділ	30.03.2026-17.04.2026	Текст розділу
	4.4 IV розділ	20.04.2026-8.05.2026	
5	Підготовка графічних матеріалів чи іншого унаочнення	11.05.2026-15.05.2026	Роздатковий матеріал, презентація
6	Оформлення кінцевого списку використаних джерел та додатків	11.05.2026-15.05.2026	Список літературних джерел
7	Оформлення та попередній захист бакалаврської роботи	18.05.2026-22.05.2026	Бакалаврська робота, рішення кафедри про допуск до захисту
8	Внесення коректив та кінцеве редагування бакалаврської роботи	25.05.2026-29.05.2026	Бакалаврська робота
9	Реєстрація бакалаврської роботи на кафедрі	1.06.2026-5.06.2026	Бакалаврська робота внесена до журналу реєстрації випускових робіт
10	Захист бакалаврської роботи	10.06.2026-28.06.2026	Рішення Екзаменаційної комісії про захист

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1 ОСНОВНІ ПОНЯТТЯ ТЕХНІЧНОЇ ДІАГНОСТИКИ.....	12
1.1 Якісні та кількісні показники надійності сучасних гаджетів.....	12
1.2 Алгоритми діагностування сучасних ІКТ.....	14
1.3 Загальні питання моделювання цифрових об'єктів діагностики.....	18
1.4 Основи верифікації вбудованих систем.....	20
Висновок до розділу.....	26
 РОЗДІЛ 2 АВТОМАТИЗАЦІЯ ОЦІНЮВАННЯ ТА ПРОГНОЗУВАННЯ ПОКАЗНИКІВ НАДІЙНОСТІ ПРОГРАМНИХ СИСТЕМ.....	28
2.1 Рекомендації вибору типу стратегії тестування на основі моделі функціонування ПС.....	28
2.2 Програмний засіб попереднього опрацювання даних про відмови програмних систем.....	34
2.3 Використання засобів автоматизації фірми ІВА для діагностики неполадок.....	37
2.4 Досвід застосування Excel та MathCad для встановлення надійності телекомунікаційних пристроїв.....	42
Висновок до розділу.....	42
 РОЗДІЛ 3 АВТОМАТИЗАЦІЯ ДІЯЛЬНОСТІ МАЛИХ ПІДПРИЄМСТВ НА ПРИКЛАДІ ДІЯЛЬНОСТІ ФІРМИ SIGMA.....	43
3.1 Перспективні напрямки автоматизації роботи малого підприємства.....	47
3.2 Автоматизація роботи підрозділу стратегічного управління.....	27
3.3 Удосконалення діяльності служби діагностики гаджетів.....	55
3.4 Застосування задач оптимізації в діяльності фірми.....	63
3.5 Технології баз даних у діяльності малого підприємства.....	69
3.6 Шляхи удосконалення запропонованих засобів автоматизації роботи малого бізнесу.....	74
Висновок до розділу.....	77
 РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ.....	78
4.1 Охорона праці при роботі за комп'ютером.....	78
4.2 Безпека в надзвичайних ситуаціях.....	83
Висновок до розділу.....	86
 ЗАГАЛЬНИЙ ВИСНОВОК.....	87
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	90

АНОТАЦІЯ

Клопот Я. С. Автоматизація процесу оцінювання та прогнозування показників надійності програмних систем. – Рукопис.

Кваліфікаційна бакалаврська робота за спеціальністю 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка». – Таврійський національний університет імені В.І. Вернадського, Київ, 2025 рік.

У бакалаврській кваліфікаційній роботі досліджено теоретичні засади технічної діагностики та автоматизованого оцінювання надійності програмних систем. Розроблено програмний засіб попереднього опрацювання даних про відмови ПС, інтегрований із системою JIRA. Запропоновано комплекс рішень для автоматизації діяльності малого підприємства на прикладі фірми Sigma, що включає розробку спеціалізованої програми для SPACE-аналізу, бази даних «Менеджер» у середовищі MS Access та інтерактивного веб-сервісу діагностики гаджетів на базі хмарних технологій.

Практичне значення полягає у можливості застосування методу нелінійного регресійного аналізу для підвищення точності прогнозування кількості помилок (на 3–5%) та скорочення тривалості тестування (в середньому на 30-40%), а також у впровадженні розроблених програмних інструментів для оптимізації стратегічних, комерційних і сервісних процесів підприємства без додаткових значних капіталовкладень.

Об'єкт дослідження – процес оцінювання, прогнозування та забезпечення показників надійності програмних систем.

Предмет дослідження – математичні моделі, методи, алгоритми та інструментальні програмні засоби для тестування, технічної діагностики та автоматизації бізнес-процесів.

Мета роботи – розробка та впровадження автоматизованих методів і програмних засобів для оцінювання надійності програмних систем, а також комплексної автоматизації діяльності структурних підрозділів малого підприємства.

Методи дослідження – системний аналіз, нелінійний регресійний аналіз, математичне та структурне моделювання, SPACE-аналіз, методи проектування реляційних баз даних.

Ключові слова: надійність програмних систем, автоматизація тестування, технічна діагностика, математичне моделювання, бази даних, автоматизація малого бізнесу, веб-сервіс, SPACE-аналіз.

ABSTRACT

Klopot Y.S. Automation of the process of evaluating and predicting reliability indicators of software systems. – Manuscript.

Qualification bachelor's thesis in specialty 174 "Automation, computer-integrated technologies and robotics." - V.I. Vernadsky Tavrichesky National University, Kyiv, 2025.

The bachelor's qualification work explores the theoretical foundations of technical diagnostics and automated reliability assessment of software systems. A software tool for pre-processing data on aircraft failures, integrated with the JIRA system, has been developed.

A set of solutions for automating the activities of a small business is proposed using the example of the Sigma company, which includes the development of a specialized program for SPACE analysis, the "Manager" database in the MS Access environment, and an interactive web service for diagnosing gadgets based on cloud technologies.

The practical significance lies in the possibility of using the nonlinear regression analysis method to increase the accuracy of forecasting the number of errors (by 3–5%) and reduce the duration of testing (on average by 30–40%), as well as in the implementation of the developed software tools to optimize the strategic, commercial and service processes of the enterprise without additional significant capital investments.

The object of research is the process of evaluating, predicting and ensuring the reliability of software systems.

The subject of research is mathematical models, methods, algorithms and software tools for testing, technical diagnostics and automation of business processes.

The purpose of the work is to develop and implement automated methods and software tools for assessing the reliability of software systems, as well as comprehensive automation of the activities of structural divisions of a small enterprise.

Research methods – systems analysis, nonlinear regression analysis, mathematical and structural modeling, SPACE analysis, methods for designing relational databases.

Keywords: reliability of software systems, testing automation, technical diagnostics, mathematical modeling, databases, small business automation, web service, SPACE analysis.

ВСТУП

Переважна більшість зразків новітньої техніки, до якої належать сучасні електронні прилади та високотехнологічні телекомунікаційні пристрої, за своєю архітектурою представляють собою складні програмно-апаратні системи (ПАС). Їхнє безперебійне функціонування базується на глибокій та неперервній взаємодії апаратних компонентів (hardware) та відповідного програмного забезпечення (software). З огляду на те, що такі комплекси дедалі частіше беруть на себе відповідальність за виконання критично важливих завдань, безпосередньо пов'язаних із гарантуванням безпеки життєдіяльності людини, процеси точного оцінювання та забезпечення заданих показників їхньої надійності набувають першочергового значення ще на етапах проектування, серійного виробництва та подальшої експлуатації.

Стрімке впровадження інноваційних технологій у сферу телекомунікацій здійснює фундаментальний вплив на архітектурну структуру та шляхи модернізації існуючих телекомунікаційних систем і мереж зв'язку. Такі трансформаційні процеси об'єктивно потребують залучення значних капітальних інвестицій, проте в результаті вони суттєво сприяють підвищенню загальної ефективності праці в найрізноманітніших галузях економіки. Головним каталізатором цих змін виступає перманентне вдосконалення мобільних терміналів, смартфонів та комп'ютерної техніки, що супроводжується невідпинним зростанням вимог з боку споживачів щодо пропускної здатності, швидкості надання та бездоганної якості інтегрованих інфокомунікаційних послуг. Саме ці фактори диктують гостру необхідність комплексної модернізації мереж, а також оптимізації їхніх топологічних структур, робочих показників та чинників, що на них впливають.

З наукової точки зору, завдання щодо удосконалення діючих інфокомунікаційних інфраструктур класифікуються як складні багатокритеріальні задачі аналізу і синтезу. Їхня надзвичайна складність обумовлюється наявністю величезної кількості взаємозалежних критеріїв

(показників) та різноманітних факторів впливу. Крім того, виникає потреба у розробці адаптивних систем управління, новітніх математичних методів розрахунку параметрів пристроїв, а також в оптимізації базових характеристик мережі в умовах її безперервної еволюції. Виходячи з цього, надзвичайно актуальною стає науково-практична задача розробки такого універсального методу, який би дозволяв із максимальною достовірністю визначати ступінь кореляційного впливу багатьох змінних факторів на конкретний цільовий показник системи. Практика показує, що традиційні методи дослідження, які найчастіше застосовуються сьогодні, нерідко виявляються недостатньо ефективними і не дають змоги сформулювати надійний та якісний прогноз щодо досягнення оптимальних показників функціонування апаратури.

У міру того, як масштаби та архітектурна складність програмно-апаратних систем загалом (і програмного забезпечення зокрема) експоненційно зростають, процедура об'єктивного і точного оцінювання показників їхньої надійності стає дедалі важчою. Саме тому на всіх без винятку етапах життєвого циклу створення складних технічних об'єктів питання всебічної верифікації виходить на передній план. Сучасні корпоративні та промислові інформаційні системи (ІС) — це високоінтегровані програмно-апаратні комплекси, життєздатність яких залежить від бездоганної та ретельної перевірки. Проте серйозною проблемою залишається те, що існуючі сьогодні методологічні підходи здебільшого є фрагментованими і не орієнтуються на синхронну, одночасну верифікацію як програмної складової, так і апаратного забезпечення всередині складної вбудованої системи.

Щоб подолати цей бар'єр та досягти повної автоматизації процесу верифікації сучасних інформаційних систем, розробникам критично необхідно спиратися на строго формалізовану математичну модель верифікації. Найбільш дієвим та поширеним інструментом реалізації такої перевірки є системне тестування, яке надає можливість здійснити глибоку та всебічну верифікацію абсолютно всіх модулів та частин системи, особливо

тих, куди вносилися будь-які архітектурні зміни чи оновлення коду. На сьогоднішній день інженерній спільноті варто приділити прискіпливу увагу розробці інноваційних методів гібридного тестування програмного та апаратного забезпечення вбудованих систем. Впровадження таких передових методів дозволить успішно автоматизувати складний рутинний процес формування наборів взаємозалежних компонент під час проведення регресійного (повторного) тестування, що, у свою чергу, гарантуватиме найвищий рівень надійності кінцевої інформаційної системи.

РОЗДІЛ 1 ОСНОВНІ ПОНЯТТЯ ТЕХНІЧНОЇ ДІАГНОСТИКИ

1.1. Якісні та кількісні показники надійності сучасних гаджетів

Сучасна наука і техніка приділяють значну увагу проектуванню та експлуатації складних телекомунікаційних і радіотехнічних комплексів, прагнучи максимально підвищити їхню ефективність. У контексті технічної діагностики новітніх гаджетів фундаментальним є поняття якості. Під якістю об'єкта розуміють комплекс його властивостей, що визначають придатність виробу виконувати своє пряме призначення та повністю відповідати критеріям, закріпленим у нормативно-технічній документації.

Ключовою складовою якості будь-якого пристрою виступає його надійність. Це здатність обладнання протягом визначеного часу стабільно підтримувати всі робочі параметри в заданих межах під час безпосереднього використання, технічного обслуговування, транспортування та зберігання. Залежно від специфіки та умов роботи, надійність є комплексною характеристикою. Вона може поєднувати в собі такі базові властивості, як безвідмовність, довговічність, придатність до ремонту та збережуваність (або певні їх комбінації).

Безвідмовність гарантує, що пристрій безперебійно виконуватиме свої функції за певних умов протягом заданого часового інтервалу або визначеного обсягу роботи, що називається наробітком. Наробіток може вимірюватися як у безперервних одиницях (наприклад, тривалість у годинах чи кілометраж пробігу), так і в дискретних величинах (кількість робочих циклів, запусків тощо). Своєю чергою, довговічність характеризує здатність техніки повноцінно працювати до моменту досягнення нею граничного стану, за умови дотримання встановлених правил технічного обслуговування та ремонту.

Граничним вважається такий стан об'єкта, коли його подальше використання стає технічно неможливим, небезпечним чи економічно недоцільним. Ознаки, що свідчать про настання цього стану, завжди чітко прописуються в проектній або нормативній документації виробу. Хоча і

безвідмовність, і довговічність пов'язані зі збереженням працездатності в процесі наробітку, між ними є суттєва різниця. Безвідмовність вимагає виключно безперервної роботи без поломок, тоді як довговічність оцінює загальний життєвий цикл пристрою, який допускає паузи на планове обслуговування та ремонтні роботи.

Важливими аспектами надійності є також ремонтпридатність та збережуваність. Ремонтпридатність визначає, наскільки конструкція об'єкта адаптована до швидкого виявлення несправностей та відновлення його повноцінної роботи. Збережуваність же відображає здатність апаратури не втрачати своїх експлуатаційних властивостей (у межах допустимих відхилень) під час або після тривалого зберігання чи транспортування. Для точного та об'єктивного оцінювання усіх перелічених властивостей на практиці використовують спеціальні кількісні показники надійності, які систематизовано в таблиці нижче.

Таблиця 1.1

Кількісні показники надійності

Властивості	Позначення показників	Найменування показників
Безвідмовність	$p(t)$ $f_T(t)$ $\bar{T}$ $\lambda(t)$ $\omega(t)$ T_B	Імовірність безвідмовної роботи Густина розподілу наробітку до відмови Середній наробіток до відмови Інтенсивність відмов Параметр потоку відмов Середній наробіток на відмову
Ремонтпридатність	$U(t)$ $f_\tau(t)$ T_B $\mu(t)$	Імовірність відновлення в заданий час Густина розподілу часу відновлення Середній час відновлення Інтенсивність відновлення
Довговічність	R_γ $t_{TCA\gamma}$ $\bar{R}$ $\bar{t}_{TCA}$	Гамма-відсотковий ресурс Гамма-відсотковий термін служби Середній ресурс Середній термін служби

Збережуваність	$\bar{t}_{збру}$ $\bar{t}_{збр}$	Гамма-відсотковий термін збереження Середній термін збереження
Комплексна властивість надійності	k_{Γ} k_{TB} $k_{OP}(t_0)$ k_{ef}	Коефіцієнт готовності Коефіцієнт технічного використання Коефіцієнт оперативної готовності Коефіцієнт збереження ефективності

Типовий графічний вигляд цих функцій проілюстровано на рисунку нижче.

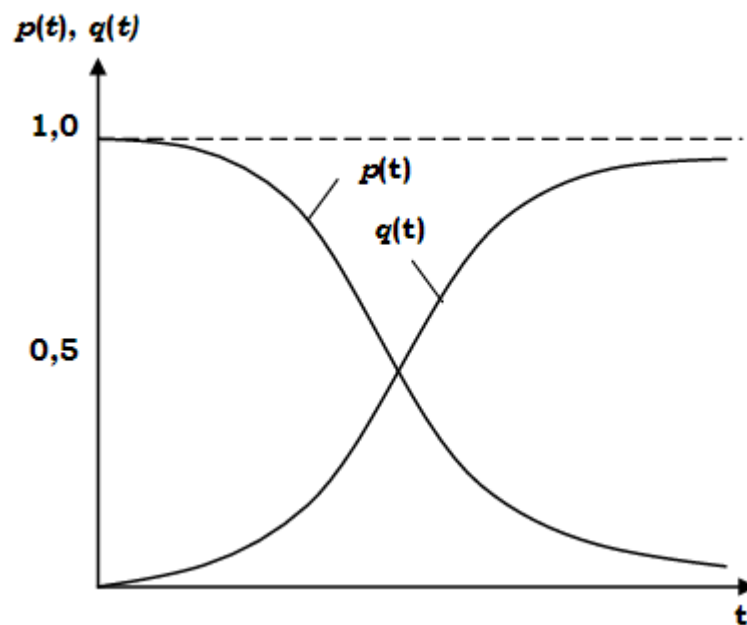


Рис. 1.1 Графічний вигляд функцій

Залежно від того, наскільки повною є вихідна інформація, розрахунок показників надійності поділяють на наближений та повний. Щоб виконати наближений розрахунок, фахівцям зазвичай вистачає загальної функціональної схеми пристрою. Натомість повний розрахунок є значно складнішим: він потребує наявності детальної принципової схеми (аж до найменших компонентів), а також точних даних про показники надійності кожного окремого елемента, реальні режими їхньої експлуатації та рівні навантажень, що на них діють.

1.2 Алгоритми діагностування сучасних ІКТ

Сучасна технічна діагностика являє собою комплексну наукову

дисципліну, фундаментальну основу якої складають різноманітні теорії, специфічні методи та інструментальні засоби, спрямовані на ефективне виявлення та точний пошук дефектів у різноманітних технічних системах. У межах цієї галузі під терміном «дефект» традиційно розуміють будь-яку, навіть найменшу, невідповідність фактичних характеристик та параметрів об'єкта тим властивостям, які були попередньо задані нормативною документацією, є необхідними для роботи або просто очікуються від пристрою (це можуть бути фізичні ушкодження, раптові відмови, системні збої тощо). Відповідно, процес виявлення дефекту зводиться до базової констатації факту: чи присутня несправність у досліджуваному об'єкті, чи вона відсутня. Натомість процедура пошуку дефекту є значно складнішою і полягає у точній просторовій та логічній локалізації місця виникнення цієї несправності всередині системи із заздалегідь визначеною похибкою або точністю.

Обидва ці взаємопов'язані процеси — виявлення наявності відхилень та безпосередній пошук місця розташування дефектів — спрямовані на комплексне визначення поточного технічного стану апаратури. У фаховій літературі вони об'єднуються під одним широким та універсальним терміном, який звучить як «технічне діагностування».

Безпосереднім об'єктом діагностування може виступати як повністю готовий до експлуатації виріб, так і його окремі конструктивні вузли, блоки, плати або навіть початкові заготовки, технічний стан яких вимагає ретельної перевірки та оцінки. У сучасній теорії технічної діагностики всю сукупність досліджуваних об'єктів прийнято чітко розділяти на два великі класи: аналогові (або неперервні) та дискретні пристрої. Функціонування аналогових об'єктів базується на сигналах у формі електричних струмів та напруг, які мають здатність набувати абсолютно будь-яких проміжних значень у межах певної заданої робочої області. До цієї категорії належать різноманітні підсилювачі частот, генератори сигналів різної форми, класичні блоки живлення та інша подібна апаратура. Що стосується дискретних об'єктів, то в них робочі сигнали здатні приймати виключно фіксовані,

дискретні значення (найчастіше це логічний «нуль» або логічна «одиниця»). Цей клас охоплює всі без винятку цифрові логічні елементи, мікросхеми та сучасні цифрові персональні електронно-обчислювальні машини (ПЕОМ). Глобальне призначення технічної діагностики полягає у гарантуванні високого рівня надійності таких систем безпосередньо на етапі їхнього практичного використання, а також у своєчасному виявленні та попередженні виробничого браку ще на конвеєрі під час виготовлення.

Якщо розглядати специфічні умови, наприклад, процес підготовки радіоелектронної техніки (РЕТ) до відповідального бойового чи оперативного застосування, то головною метою здійснення технічного діагностування стає жорсткий контроль працездатності. У таких критичних ситуаціях виділяють лише два можливі технічні стани об'єкта: він є або повністю працездатним і готовим до виконання завдань, або критично непрацездатним.

З огляду на це, ключовим та найбільш відповідальним завданням будь-якого процесу діагностування є глибока перевірка загальної справності обладнання, підтвердження його повної працездатності та правильності виконання закладених алгоритмів функціонування. Паралельно з цим обов'язково вирішується завдання пошуку та ідентифікації тих прихованих чи явних дефектів, які здатні порушити нормальний робочий стан, погіршити справність або спотворити правильність роботи системи.

Для практичної реалізації процедури діагностування технічного стану будь-якого класу об'єктів залучається широкий арсенал спеціалізованих засобів. Їх можна умовно поділити на апаратні (фізичні прилади, осцилографи, стенди), суто програмні (діагностичні утиліти, тестові коди) або ж комбіновані програмно-апаратні комплекси, взаємодію яких проілюстровано на рисунку нижче.



Рис. 1.2 Класифікація засобів діагностування

Крім того, не варто забувати, що в ролі своєрідного та дуже гнучкого засобу діагностування може виступати і висококваліфікований людський фактор — безпосередньо оператор установки, контролер відділу технічного контролю (ВТК) або сервісний інженер-налагоджувальник, який візуально або за допомогою слуху аналізує поведінку системи на основі свого багаторічного досвіду.

З математичної та логічної точок зору, алгоритм технічного діагностування (АТД) у своєму найзагальнішому вигляді являє собою чітко структуровану сукупність так званих елементарних (базових) перевірок досліджуваного об'єкта. Цей алгоритм обов'язково супроводжується набором суворих правил, які однозначно регламентують послідовність та порядок виконання цих перевірок у часі, а також містить критерії та правила для подальшого аналізу й правильної інтерпретації отриманих результатів вимірювань.

Уся ця логічно вибудована та впорядкована послідовність елементарних кроків перевірки формує єдиний інструмент — тест діагностування. За своїм функціональним призначенням такі тести поділяються на дві основні

категорії. Тест перевірки розробляється спеціально для швидкого підтвердження факту повної справності або часткової працездатності об'єкта діагностування. Натомість тест пошуку дефекту має значно складнішу та глибшу структуру; він призначений для максимально точного визначення просторового місця виникнення несправності, а також, за наявності такої необхідності в інженерів, для детального з'ясування першопричини та конкретного виду виявленого дефекту.

1.3 Загальні питання моделювання цифрових об'єктів діагностики

У сучасній теорії технічної діагностики всі математичні моделі, що застосовуються для опису та аналізу цифрових пристроїв, фундаментально поділяються на дві великі категорії: функціональні та структурні. Якщо функціональні моделі зосереджені переважно на описі алгоритмів роботи пристрою як «чорного ящика», то структурні моделі мають принципово інше призначення та рівень деталізації.

Структурні моделі представляють собою детальні логічні схеми, які чітко задаються через вичерпний перелік усіх наявних входів, виходів, внутрішніх базових елементів об'єкта діагностики, а також повної топології зв'язків між ними. Їхня головна особливість полягає в тому, що вони орієнтовані не просто на абстрактну перевірку загальних функцій, які має виконувати пристрій, а на цілеспрямований пошук та ідентифікацію фізичних чи логічних несправностей безпосередньо в елементах його апаратної структури. Під час виконання структурного моделювання перед інженерами постають два ключові завдання: адекватне моделювання самих компонентів, обраних із заданого логічного базису, та правильний вибір математичної моделі для опису сигналів або змінних (які, своєю чергою, класифікуються на входні, вихідні та внутрішні проміжні вузли).

Наочний приклад типової комбінаційної логічної мережі, стан якої визначається виключно комбінацією поточних значень на входах, наведено на рисунку нижче. На цій принциповій схемі латинськими літерами *a*, *b*, *c*, *d* умовно позначені незалежні входні змінні (або стимули), тоді як символ *z*

відповідає за результуючу вихідну функцію (відгук системи).

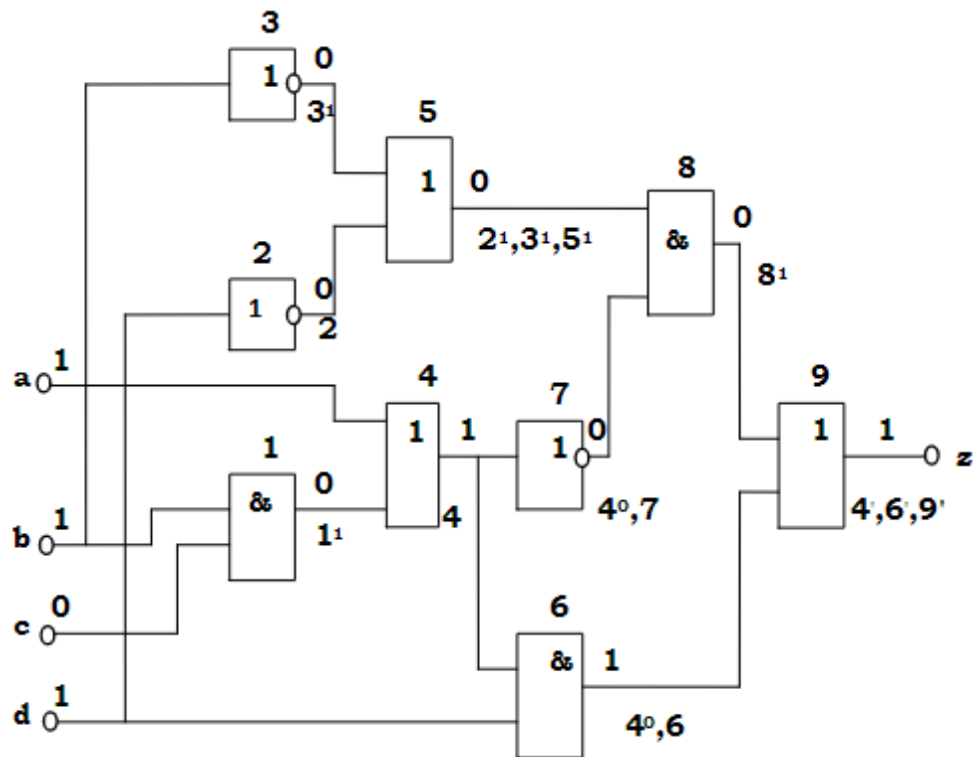


Рис. 1.3 Комбінаційна логічна мережа

Під час безпосереднього процесу моделювання кожній вхідній, вихідній та внутрішній змінній обов'язково ставиться у відповідність певний сигнал, що обирається з так званого алфавіту моделювання. Історично найпростішим, базовим та найбільш розповсюдженим варіантом є класичний двійковий (бінарний) алфавіт, який оперує лише двома дискретними станами: 0 та 1. Однак на практиці, особливо під час аналізу перехідних процесів чи збоїв, цього часто буває недостатньо.

Саме тому в сучасних системах автоматизованого проектування надзвичайно широкого поширення набуло трійкове моделювання. Воно використовує розширений трійковий алфавіт (0, 1, x), де додатковий символ x відіграє критично важливу роль. Він означає, що реальний логічний сигнал на певній лінії в даний момент часу є невідомим, невизначеним або перебуває у стані переходу (тобто це може бути як логічна одиниця, так і логічний нуль). Більше того, у загальному випадку, для забезпечення максимальної повноти та достовірності результатів технічної діагностики складних цифрових систем, може успішно застосовуватися спеціальне багатозначне моделювання, алфавіт якого налічує від 4 до 10 різних

символів-станів [8].

1.4 Основи верифікації вбудованих систем

Фундаментальною передумовою для глибокого розуміння сутності проблем надійності засобів зв'язку є детальне вивчення основ їхньої експлуатації. Лише спираючись на ці знання, інженери здатні обґрунтовано підходити до вибору найефективніших шляхів та інструментальних методів, спрямованих на підвищення експлуатаційної безвідмовності апаратури. Сучасна інженерна практика виділяє кілька магістральних напрямів верифікації: різноманітні види експертиз, методи статичного аналізу, строгі формальні методи, динамічне тестування, а також комплексні синтетичні підходи (рис. 1.1) [13].

Якщо розглядати категорію експертиз, то традиційно до неї відносять організаційні перевірки, суто технічні експертизи, процедури наскрізного контролю, формальні інспекції та незалежні аудити. Починаючи з середини 1990-х років, у світовій індустрії також активно впроваджуються методи сценарного оцінювання архітектури програмного забезпечення. Хоча ці методи є потужним аналітичним інструментом, їх зазвичай відокремлюють від класичних форматів експертизи. Головна унікальна риса будь-якої експертизи, що відрізняє її від решти підходів до верифікації, полягає у можливості працювати безпосередньо з оригінальними артефактами життєвого циклу розробки. Тобто для її проведення не потрібно будувати складні математичні моделі (як того вимагають формальні методи) або очікувати на результати компіляції та виконання коду (що є обов'язковим для динамічних перевірок).

Експертиза є надзвичайно гнучким інструментом, який можна застосовувати до будь-яких характеристик програмного забезпечення та на будь-якій стадії його створення — від збору вимог до релізу. Вона дає змогу ідентифікувати величезний спектр потенційних помилок ще на етапі первинного проектування. Такий проактивний підхід кардинально скорочує час «життя» дефекту в системі та нівелює його руйнівні наслідки для якості

всіх наступних похідних компонентів системи [7]. Проте варто пам'ятати, що результативність експертизи критично залежить від людського фактора: рівня компетенції фахівців, їхньої особистої мотивації, грамотної організації робочого процесу та якості внутрішніх комунікацій. Ігнорування цих аспектів керівництвом може спровокувати неефективний розподіл проектних ресурсів та міжособистісні конфлікти всередині команди розробників.

Кожен формат проведення експертизи характеризується власною специфікою та цільовим призначенням:

- **Технічна експертиза:** являє собою глибокий і систематичний аналіз проектних артефактів, який виконується групою висококваліфікованих інженерів. Головна мета — оцінити внутрішню логічну узгодженість рішень, їхню технічну точність, повноту, а також відповідність галузевим стандартам, внутрішнім корпоративним процесам та глобальним цілям проекту.

- **Наскрізний контроль:** інтерактивний метод колективного аналізу, під час якого автор або відповідальний член команди покроково презентує колегам усі характеристики створеного артефакту. Учасники зустрічі в режимі реального часу аналізують матеріал, ставлять уточнюючі запитання, генерують зауваження та фіксують можливі логічні хиби чи відхилення від стандартів.

- **Інспекція:** суворо регламентоване та послідовне дослідження характеристик продукту. Воно завжди базується на заздалегідь затвердженому плані (чек-листі) і має єдину прагматичну мету — цілеспрямований пошук та документування аномалій, помилок і дефектів.

- **Аудит:** зовнішній зріз якості. Це аналіз процесів життєвого циклу та проектних артефактів, який принципово виконується незалежними фахівцями, що не є частиною команди розробників. Аудитори перевіряють, наскільки реальний стан справ відповідає юридичним контрактам, базовим завданням проекту та міжнародним стандартам якості.

На відміну від експертиз, статичний аналіз орієнтований на автоматизовану перевірку суворих формалізованих правил написання коду

або побудови архітектурних артефактів інформаційних систем (ІС). Цей метод шукає вразливості та помилки шляхом зіставлення коду із відомими шаблонами дефектів. Статичний аналіз чудово піддається автоматизації та переважно делегується спеціалізованому програмному забезпеченню (хоча налаштування правил та стандартів кодування іноді потребує ручного втручання). Незважаючи на зручність, цей підхід має обмеження: він застосовується лише до сирцевого коду або специфічних форматів даних і здатен розпізнавати лише лімітований набір типів помилок. Головною проблемою строгого статичного аналізу є високий відсоток хибних спрацьовувань (велика надмірність попереджень), або ж, навпаки, ризик пропустити критичний баг під час використання менш агресивних алгоритмів перевірки. Проте завдяки своїй доступності інструменти автоматичного статичного аналізу стали індустріальним стандартом. Багато успішних технік такої перевірки з часом інтегруються безпосередньо в компілятори або стають невід'ємною частиною семантичних правил сучасних мов програмування.

Цікаво, що на практиці строгі формальні методи верифікації значно частіше і успішніше застосовуються під час розробки апаратного забезпечення, ніж програмного. Ця тенденція пояснюється колосальною вартістю виправлення помилок у «кремнії», високою структурною однорідністю мікросхем, можливістю масового повторного використання топологічних блоків, а також звичкою інженерів-апаратників працювати в умовах жорстких фізичних обмежень та математично точних специфікацій.

У контексті формальних методів виділяють логіко-алгебраїчні моделі, які також називають логічними обчисленнями. Під час моделювання ПЗ такі конструкції описують певний спектр властивостей системи, що еволюціонують у часі. Однак вони лише констатують факт зміни, не розкриваючи внутрішніх механізмів того, завдяки чому саме ці властивості трансформуються.

На противагу їм існують здійснимі, або операційні моделі. Їхня ключова перевага полягає у здатності бути фізично «виконаними» або

симульованими. Це дозволяє інженерам у динаміці відстежувати, як саме змінюються стани та властивості модельованого програмного забезпечення. По суті, кожна така модель функціонує як специфічна програма, запущена в ізольованому середовищі суворо визначеної віртуальної машини. З математичної точки зору, всі варіації здійснених моделей є логічним розвитком та узагальненням класичної теорії скінченних автоматів.

Процедура перевірки моделей — це потужний автоматизований метод доведення того, що певна здійснима модель (яка відображає архітектурне рішення) гарантовано задовольняє набір вимог, записаних мовою логіко-алгебраїчних формул. Найчастіше для опису таких вимог застосовують апарат темпоральної (часової) логіки. У ролі самої моделі зазвичай виступає складна структура — скінченний автомат, стани якого відповідають логічним значенням перевірюваних формул (у науковій літературі це відомо як модель Кріпке). Спеціалізований програмний інструмент пропускає через себе цю модель і видає один із трьох результатів: стовідсотково підтверджує наявність заданих властивостей, генерує контрприклад (сценарій збою, що призводить до порушення вимог), або ж зупиняє роботу без вердикту через вичерпання обчислювальних ресурсів (проблема вибуху станів).

Ще одним різновидом формального аналізу є перевірка узгодженості. Цей метод передбачає порівняльний аналіз двох здійснених моделей. Перша модель репрезентує сам артефакт (реальний проект або прототип компонента), а друга виступає еталоном, що містить ідеальні вимоги до поведінки системи. Еталонна модель найчастіше формалізується у вигляді узагальнених автоматів, систем марковських переходів або мереж Петрі.

Динамічні методи верифікації кардинально відрізняються тим, що оцінка властивостей програмної системи базується виключно на аналізі її реальної фізичної роботи в середовищі виконання (або роботи її високоточних прототипів). Найвідомішими прикладами цього класу є класичне тестування, імітаційне моделювання та runtime-моніторинг. Оскільки для їх реалізації обов'язково потрібен готовий код, що компілюється та виконується, ці методи технічно неможливо застосувати на

ранніх етапах збору аналітики чи абстрактного проектування. Їхня суперсила — контроль поведінки ІС у реальному або наближеному до бойового оточенні. Динамічна верифікація здатна виявити специфічні дефекти інтеграції та збої пам'яті, які проявляються виключно в момент роботи програми.

Проте застосування динамічних методів вимагає залучення фахівців із високою інженерною кваліфікацією. Необхідно розробити грамотну архітектуру тестів, створити надійне тестове оточення (фреймворк) та розгорнути системи моніторингу метрик. Хоча базова інфраструктура для тестування створюється один раз і може масштабуватися на різні проекти, самі тестові сценарії доводиться писати з нуля для кожної нової ІС. Разом з тим, процес написання тестів на ранніх стадіях має колосальний побічний ефект: він змушує інженерів-тестувальників глибоко аналізувати вимоги. Таким чином, розробка тестів перетворюється на неформальну, але дуже прискіпливу експертизу проектної документації, виявляючи логічні прогалини в ТЗ ще до написання першого рядка коду.

Формування вичерпних тестових наборів, здатних дати об'єктивну оцінку якості складної корпоративної системи, є надзвичайно ресурсоемною задачею. Через високу вартість повноцінного покриття коду, індустрія часто вдається до компромісних, менш надійних, але економічно вигідних технік. До них належить ймовірнісне тестування, де вхідні параметри генеруються випадковим чином, або ж тестування виключно «щасливих шляхів», яке покриває тільки базові та найпростіші сценарії експлуатації продукту.

З метою подолання недоліків ізольованих підходів, у сучасній інженерії виокремилися синтетичні (гібридні) методи, які інтегрують елементи формальної логіки у динамічне тестування. До цієї категорії відносять тестування на основі моделей та моніторинг формальних властивостей у реальному часі. Передові інструменти генерації тестів сьогодні активно комбінують математичну специфікацію вимог із глибоким статичним аналізом вихідного коду. Головна філософія синтетичних методів — синергія переваг різних підходів для взаємної компенсації їхніх слабких сторін.

Тестування на основі моделей використовує математично точні моделі вимог до ІС як фундамент для автоматичної генерації тестових сценаріїв. Інформація, закладена в таких моделях, слугує базисом для розрахунку критеріїв повноти покриття. Згенеровані у такий спосіб тести абстраговані від специфіки реалізації конкретного коду, але при цьому гарантують репрезентативну вибірку всіх можливих робочих станів з точки зору початкових бізнес-вимог.

На сьогодні індустріальні методи тестування на основі моделей спираються на перевірку узгодженості автоматів і систем логічних переходів. Залежно від обраного математичного апарату, вони поділяються на три основні категорії:

1. Методи на базі скінченних автоматів: це історично найбільш досліджений напрямок. Для них існують чіткі математичні гарантії повноти перевірок та відомі всі граничні обмеження.

2. Методи з використанням розширених автоматів: вони концептуально зводяться до класичних автоматів, проте оперують значно складнішими критеріями тестового покриття, оскільки враховують потоки даних та змінні стани всередині самої розширеної моделі.

3. Методи на основі систем переходів: цей підхід є незамінним під час тестування складних децентралізованих та розподілених систем, оскільки спроба описати їх через звичайні скінченні автомати призводить до колапсу через експоненційне зростання кількості станів. Хоча ці методи рідко забезпечують 100% гарантію повноти, сучасний інструментарій успішно застосовує інтелектуальні евристики для формування достатніх та ефективних тестових наборів.

Нарешті, існують методи побудови тестів, що базуються безпосередньо на формальному аналізі властивостей ПЗ. Вони використовують апарат математичної логіки для точної класифікації ситуацій та цілеспрямованої генерації вхідних даних. У рамках парадигми *model checking*, тестові сценарії формуються як еталонні представники класів еквівалентності. Практично це реалізується так: інструмент намагається довести недосяжність певної

ділянки коду, і згенерований ним контрприклад (шлях, який спростовує цю недосяжність) автоматично перетворюється на готовий тестовий сценарій. Своєю чергою, методи на основі дедуктивного аналізу генерують тестові дані, виходячи з аналізу специфічних крайових умов та особливих математичних випадків, виявлених під час логічного виведення властивостей системи.

Висновок до розділу

На основі проведеного аналізу теоретичних засад технічної діагностики та верифікації сучасних інформаційно-комунікаційних і вбудованих систем можна стверджувати, що ефективність експлуатації складних технічних систем фундаментально залежить від їхньої надійності. Вона є комплексною характеристикою, яка об'єднує властивості безвідмовності, довговічності, ремонтпридатності та збережуваності. Об'єктивна оцінка технічного стану вимагає застосування системи точних кількісних показників, вибір яких напряму залежить від повноти вихідних даних про об'єкт. Своєю чергою, виявлення та просторова локалізація дефектів потребують розробки алгоритмів технічного діагностування, які оперують спеціальними тестами перевірки та пошуку. Залежно від природи сигналів (аналогові чи дискретні), ефективний контроль працездатності досягається лише за рахунок синергії апаратних, програмних засобів та кваліфікованого людського фактора.

Для точної ідентифікації несправностей на рівні окремих компонентів мікроелектроніки критично важливою є побудова структурних логічних моделей. Перехід від класичного двійкового до багатозначного (зокрема, трійкового) алфавіту моделювання дає змогу адекватно описувати невизначені та перехідні стани, значно підвищуючи достовірність діагностики цифрових пристроїв. Поряд із цим, гарантування коректної роботи сучасних апаратно-програмних комплексів вимагає застосування багаторівневої верифікації. Максимальний рівень безвідмовності досягається шляхом комбінування експертиз на ранніх стадіях проектування, статичного

аналізу коду, суворих формальних методів (зокрема, model checking для апаратної частини) та динамічного тестування.

Найбільш перспективним напрямом розвитку інженерії на сьогодні є застосування синтетичних методів, таких як тестування на основі моделей (model-based testing). Інтеграція математичної специфікації вимог із динамічними перевітками дозволяє збалансувати недоліки ізольованих підходів, оптимізувати ресурси на тестування та забезпечити високу репрезентативність покриття. Загалом, інтеграція методів технічної діагностики із сучасними практиками багаторівневої верифікації формує надійний науково-прикладний фундамент для створення та експлуатації складних автоматизованих систем. Це дозволяє мінімізувати ризики критичних відмов та гарантувати безперебійне виконання обладнанням його цільових функцій.

2. АВТОМАТИЗАЦІЯ ОЦІНЮВАННЯ ТА ПРОГНОЗУВАННЯ ПОКАЗНИКІВ НАДІЙНОСТІ ПРОГРАМНИХ СИСТЕМ

2.1. Рекомендації щодо вибору типу стратегії тестування на основі моделі функціонування ІС

Детальний аналіз сучасної фахової літератури, а також результати проведених практичних досліджень дозволили сформулювати чіткий алгоритм дій. Спираючись на побудовану модель функціонування програмних систем (ПС) та емпіричні дані, отримані в процесі тестування за допомогою методів автоматизованого функціонального тестування (АФСТ), було розроблено комплексний перелік рекомендацій. Вони спрямовані на оптимізацію процесу вибору стратегії тестування безпосередньо під час виконання проектів із розробки програмного забезпечення. Ці рекомендації зводяться до наступних ключових етапів:

1. Вибір оптимального типу модуля на етапі декомпозиції ПС для побудови моделі її функціонування.

Під час проведення попереднього аналізу програмної системи категорично не рекомендується обирати за базову одиницю такий тип модуля (наприклад, окремий пакет, клас або дрібний метод), загальна кількість яких у системі перевищує 1000 одиниць. Нехтування цим правилом призведе до того, що тривалість аналітичних процедур перевищить усі допустимі часові норми, виділені на етап попереднього аналізу архітектури. Для точного математичного визначення необхідної кількості модулів та обсягів тестування пропонується використовувати наступні розрахункові формули:

$$S_f = \prod_{i=1}^{N_f} \prod_{v \in V_i} \prod_{used \cup V_i changed} card(EC) \quad (2.1)$$

де N_f — загальна кількість вхідних змінних конкретного методу;

EC — множина виділених класів еквівалентності для відповідної змінної v .

Виходячи з цього, загальну кількість можливих тестових сценаріїв для об'єкта рівня «клас» запропоновано обчислювати як сумарну кількість сценаріїв тестування для кожної його внутрішньої функції або методу:

$$S_0 = \sum_{f=1}^{N_M} S_f \quad (2.2)$$

де N_M — загальна кількість методів, що входять до складу аналізованого класу.

Своєю чергою, сукупна кількість тестових сценаріїв, яка є необхідною та достатньою для стовідсоткового покриття цілих пакетів програмної системи, розраховується як сума необхідних сценаріїв для кожного окремого класу, що входить до складу цього пакету. Це описується наступним співвідношенням:

$$S_p = \sum_{0=1}^{N_0} S_0 \quad (2.3)$$

де N_0 — кількість створених екземплярів (об'єктів) усіх класів безпосередньо під час виконання програми (в режимі runtime);

S_0 — базовий параметр, що визначається за попередньою формулою (2.2).

Для підтвердження цих теоретичних викладок у дослідженні О.О.

Нитребич було проведено серію експериментів. Тестування виконувалося на апаратній платформі з такими технічними характеристиками: процесор Intel Atom N2800 (з тактовою частотою 1.86 GHz), обсяг оперативної пам'яті 2 Гб, операційна система 32-розрядної архітектури. Результати цього дослідження, які яскраво демонструють залежність витраченого часу на аналіз моделі функціонування ПС від загальної кількості її модулів, наведено на графіку нижче.

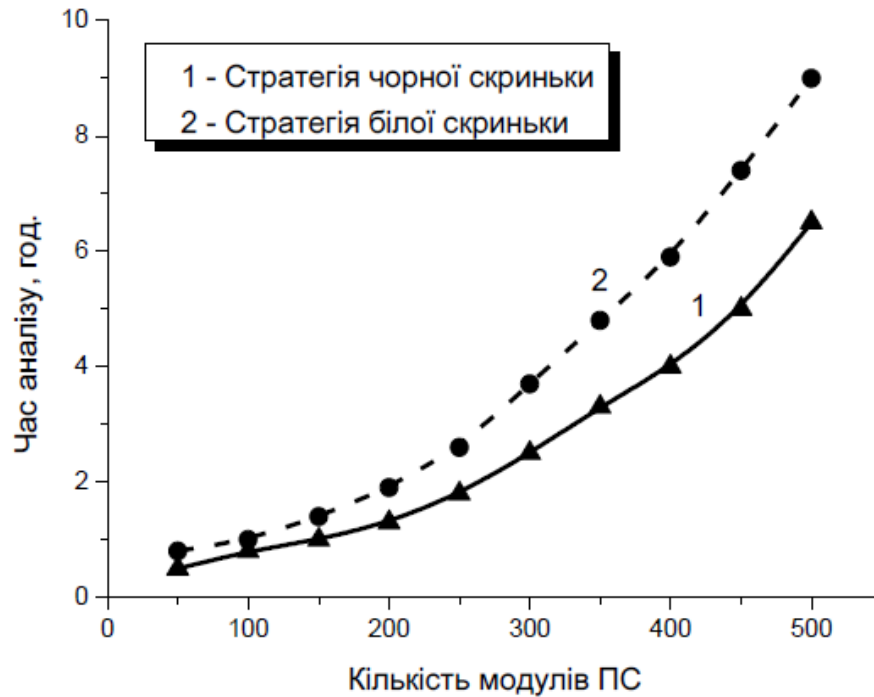


Рис. 2.1 Графік залежності часу попереднього аналізу ПС від кількості модулів для двох стратегій тестування

2. Ідентифікація параметрів моделі функціонування ПС шляхом глибокого аналізу вихідного коду.

До переліку критично важливих метрик архітектурної моделі належать: загальна кількість виділених модулів, сумарна кількість змінних у програмному коді системи, а також коефіцієнт зв'язності між цими модулями. Збір та обчислення цих параметрів здійснюється автоматизовано, за допомогою інструментів глибокого синтаксичного аналізу (парсингу) сирцевого програмного коду.

3. Вибір цільової стратегії тестування з огляду на середню кількість задіяних змінних.

Спираючись на аналіз отриманих емпіричних залежностей (проілюстрованих на рис. 2.1 та рис. 2.2), можна з упевненістю констатувати, що фундаментальним критерієм для генерації ефективного набору тестів є саме середня кількість змінних у межах одного модуля, а також усереднений показник зв'язності. Отримані результати дозволяють сформулювати такі практичні висновки:

- Як наочно демонструє рис. 2.2, у ситуаціях, коли середня кількість змінних у модулі перевищує показник у 30 одиниць, використання виключно стратегії «чорної скриньки» стає малоефективним. Вона мінімально впливає на достовірність підсумкового оцінювання надійності ПС і в таких випадках повинна застосовуватися лише для базового покриття головних користувацьких сценаріїв та перевірки критичного функціоналу.

- Натомість, якщо середня кількість змінних у модулі ПС є невеликою і не перевищує 7 одиниць, стратегія тестування «чорної скриньки» демонструє високу ефективність. У таких умовах від повноцінного тестування за стратегією «білої скриньки» можна майже повністю відмовитися (обмежившись лише написанням стандартних unit-тестів на рівні розробників). Це пояснюється тим, що фінансові та часові витрати на розгортання складного середовища тестування і написання детальних тестів будуть економічно та технічно невиправданими.

- У проміжних випадках, коли середня кількість змінних становить від 7 до 30 одиниць, система потребує більш глибокого дослідження. Необхідно проводити додатковий аналіз інших метрик коду, а максимально ефективне тестування обов'язково матиме комбінований характер (гібридне застосування обох стратегій).

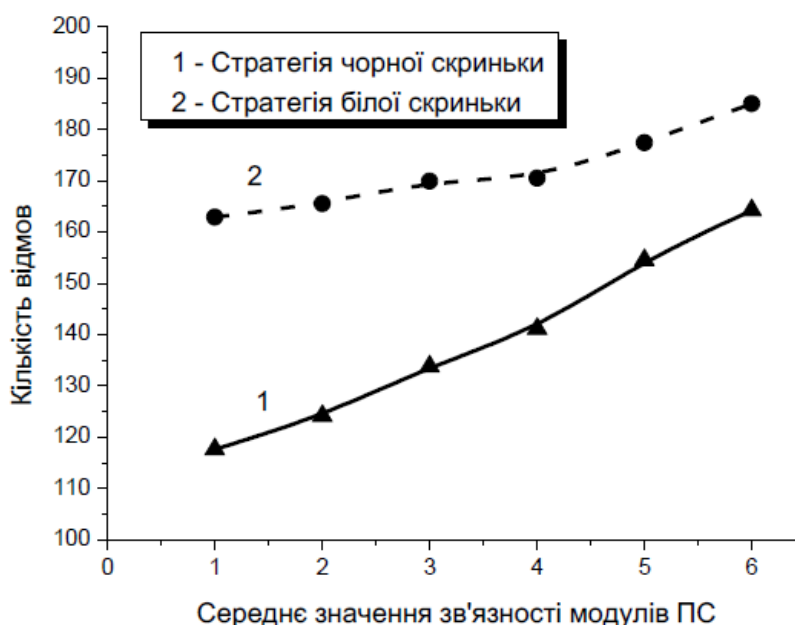


Рис. 2.2. Графік залежності кількості виявлених під час тестування відмов від зв'язності модулів ПС для двох стратегій тестування

4. Визначення характеристик помилок моделі на основі результатів попереднього тестування ПС.

До категорії прихованих (латентних) характеристик моделі функціонування системи прийнято відносити внутрішню складність самих помилок та складність процедури їхньої ідентифікації. Оцінити ці параметри можна опосередковано — проаналізувавши частоту виникнення реальних відмов у системі та зіставивши її з цикломатичною складністю відповідних ділянок коду.

Перед початком основної фази тестування сирцевий код піддається попередньому опрацюванню. Мета цього етапу — генерація лог-файлів (протоколів), які містять детальну інформацію про всі фактичні переходи управління між різними модулями ПС та логують зафіксовані системою відмови. Спираючись на ці дані, QA-інженери розраховують частоту прояву відмов та класифікують помилки за рівнем складності. Крім того, виявлені потенційні відмови (наприклад, пов'язані з некоректними значеннями змінних, арифметичними переповненнями, помилками ініціалізації пам'яті тощо) доцільно інтегрувати безпосередньо в модель функціонування ПС ще на етапі статичного аналізу коду.

5. Особливості застосування та аналіз стратегії тестування «чорної скриньки».

Зважаючи на те, що на практиці переважна більшість процесів забезпечення якості охоплює одночасне використання стратегій як «білої», так і «чорної» скриньки, критично важливо правильно розподілити ресурси та визначити оптимальну кількість тестових випадків для кожної з них. Багаторічний емпіричний досвід інженерів із тестування ПС свідчить, що процеси проектування, підготовки та безпосереднього виконання тестів за принципом «чорної скриньки» є приблизно у 10–20 разів дешевшими та швидшими порівняно зі стратегією «білої скриньки». Проте варто розуміти, що такий підхід є менш ефективним для пошуку глибоких архітектурних дефектів.

Враховуючи цей економічний фактор, класичний цикл верифікації

зазвичай починають саме з масованого тестування за стратегією «чорної скриньки», і лише після цього переходять до «білої». Така послідовність дозволяє швидко та з мінімальними витратами виявити основний масив поверхневих відмов. Складніші ж, специфічні та важкодоступні баги залишаються для виявлення на етапі «білої скриньки». Цілком логічно, що тестування «чорною скринькою» слід продовжувати рівно до того моменту, поки показник частоти виявлення нових дефектів залишається на економічно виправданому та високому рівні.

Для того, щоб отримати чіткий математичний критерій переходу від стратегії «чорної скриньки» до етапу «білої скриньки», було проведено дослідження програмних систем із середнім розміром у $N=1000$ модулів. Щоб врахувати вплив кількості логічних вузлів та внутрішньої складності помилок, у фаховій літературі [6] пропонується використовувати таку формулу для обчислення граничної кількості знайдених відмов, після досягнення якої рекомендовано змінювати підхід на «білу скриньку»:

$$\lambda < \frac{\lambda_0}{w_{WB}} \frac{\ln(N)}{\ln(1000)f_e} \quad (2.5)$$

де базовими змінними у цій закономірності виступають:

λ_0 - початкова інтенсивність відмов програмного забезпечення;

w_{WB} - відносна складність формування тестів за стратегією «білої скриньки» у порівнянні з розробкою тестів для «чорної скриньки» (цей параметр розраховується суто емпірично та є унікальним для кожної досліджуваної ПС);

f_e - а також загальна математична складність самих помилок.

6. Критерії завершення тестування за стратегією «білої скриньки».

Складність наявних помилок є фундаментальним фактором, що безпосередньо впливає на підсумкову надійність будь-якої програмної системи. Аби гарантувати замовнику справді високий ступінь надійності,

особливо за наявності прихованих відмов, спрацювання яких залежить від специфічної комбінації кількох змінних та перетину їхніх класів еквівалентності, рішення про завершення етапу «білої скриньки» слід приймати зважено. Його рекомендується ухвалювати з урахуванням параметра α — процентного відношення тих відмов, виникнення яких залежить від взаємодії трьох та більше змінних (що є індикатором високої складності помилок):

- Якщо параметр $\alpha > 1\%$, команді наполегливо рекомендується продовжити процес перевірки та провести додатковий глибокий аналіз коду з метою ідентифікації прихованих та складних відмов.

- Якщо ж параметр $\alpha < 1\%$, стадію «білої скриньки» можна завершувати та повертатися до регресійного тестування за стратегією «чорної скриньки».

Такий суворий критерій зумовлений тим, що помилки множинної залежності є надзвичайно критичними для стабільності системи, а їхній пошук стандартними методами є вкрай складним. Ці дефекти можуть бути локалізовані та усунені виключно під час глибокого тестування за стратегією «білої скриньки», коли всі інші, так звані «прості» та поверхневі баги, вже були успішно виявлені та виправлені розробниками.

2.2. Програмний засіб попереднього опрацювання даних про відмови програмних систем

У сучасній інженерній практиці існують задокументовані відомості щодо успішної реалізації та активного застосування спеціалізованого програмного інструментарію, призначеного для попереднього опрацювання статистичних даних про відмови програмних систем (ПС). Ця структурована інформація слугує критично важливою вхідною базою для різноманітних математичних моделей надійності типу «чорної скриньки», зокрема для моделі надійності, що оперує специфічним показником складності. Крім того, такі підготовлені дані активно залучаються для точного розрахунку показників надійності окремих структурних модулів ПС у тих випадках, коли інженери спираються на багаторівневі архітектурні моделі. Головне

призначення цього програмного засобу полягає у систематизації та глибокому попередньому опрацюванні масивів даних тестування, які автоматично вивантажуються з популярної системи відстеження помилок Atlassian JIRA. Кінцевою метою цих маніпуляцій є забезпечення максимально точного та достовірного оцінювання показників надійності ПС за допомогою відповідного математичного апарату.

Щодо ключових експлуатаційних та функціональних можливостей розробленого програмного засобу, то вони охоплюють надзвичайно широкий спектр аналітичних завдань. Передусім інструмент забезпечує повністю автоматизоване отримання первинних результатів тестування безпосередньо із серверів JIRA (наприклад, із відкритих ресурсів на кшталт <https://jira.springsource.org>). Наступним етапом є комплексне опрацювання отриманих результатів, яке включає створення нових та видалення неактуальних записів про виявлені відмови. Особливу увагу розробники приділили аналітичному ядру: система здатна здійснювати нормування результатів залежно від кількості залучених тестувальників, виконувати точне розбиття загальної статистики відмов на чітко задані часові інтервали, а також проводити сегментоване опрацювання статистичних даних із розрізненням типів дефектів (відстежуючи критичні, блокуючі, тривіальні та інші помилки). Додатково реалізовано складний алгоритм обчислення кумулятивної (накопиченої) кількості відмов.

Окрім суто обчислювальних функцій, система має потужний модуль для формування звітної документації. Користувачам доступні опції гнучкого сортування та багатокритеріального фільтрування під час відображення інтерактивних таблиць із результатами відмов ПС. Для кращого візуального сприйняття інструмент автоматично генерує графічні залежності динаміки відмов у часі. Усі сформовані аналітичні звіти можна не лише вивести на екран монітора, але й зручно зберегти у вигляді зовнішніх файлів для подальшого використання у документації проекту. З технічної точки зору, цей програмний засіб спроектовано та написано об'єктно-орієнтованою мовою програмування Java, а безпосередній процес написання коду

здійснювався у професійному інтегрованому середовищі розробки Eclipse Kepler. Для надійної, швидкої та безпечної інтеграції з екосистемою JIRA розробники успішно застосували стандартний архітектурний стиль взаємодії JIRA REST API.

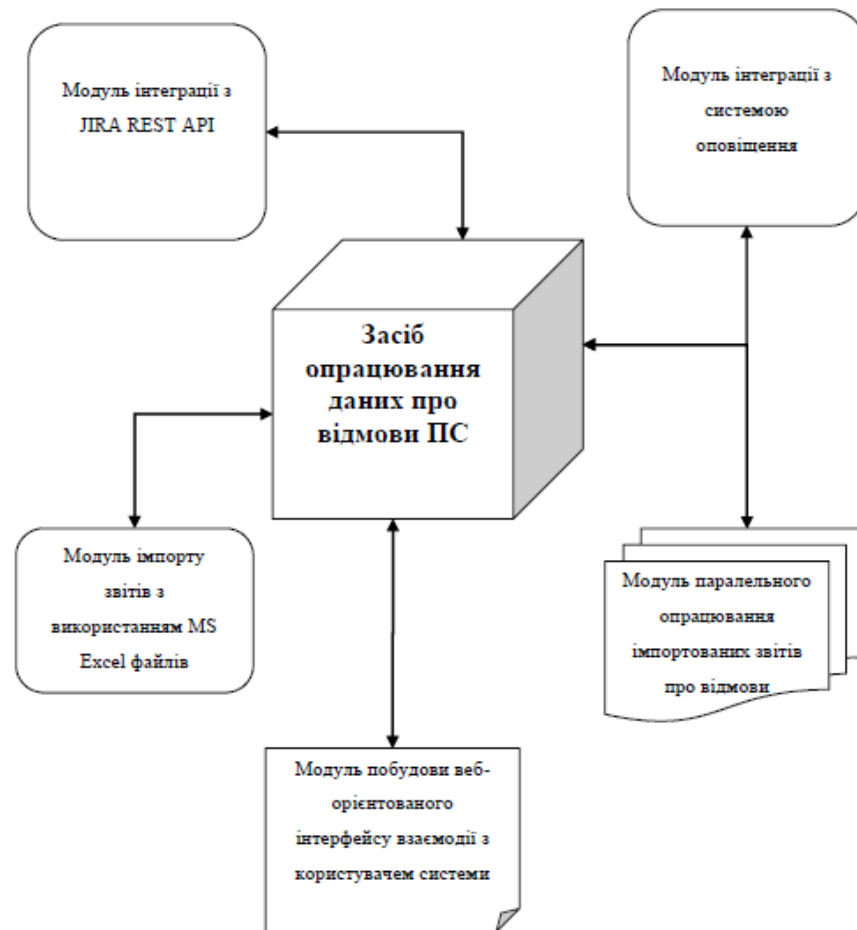


Рис. 2.3 Загальна архітектура засобу опрацювання даних про відмови ПС

Відповідно до суворих технічних вимог, висунутих на етапі проектування, було розроблено багаторівневу модульну архітектуру програмного засобу, що відповідає за безшовну інтеграцію із зовнішньою системою обліку виявлених відмов. Детальну структурну схему цієї архітектури наочно зображено на рисунку вище. Фундаментом цієї конфігурації виступає базовий модуль аналізу запитів користувача, керування процесом імпортування та інтеграції із системою обліку помилок. Він відіграє роль своєрідного системного ядра (головного модуля), яке бере на себе функції диспетчера: суворо контролює всі інтеграційні процеси та забезпечує стійку взаємодію між усіма іншими складовими компонентами

системи.

Іншим критично важливим вузлом є модуль інтеграції з JIRA REST API, який спеціалізується виключно на обміні даними з платформою Atlassian JIRA. Він бере на себе відповідальність за безпечну аутентифікацію та авторизацію користувачів у системі, коректне формування мережових запитів, їх первинну обробку та подальше конвертування отриманих текстових даних із формату JSON у повноцінні об'єкти мови Java. Паралельно з цим працює спеціальний модуль інтеграції із системою оповіщення. Його головним завданням є підключення до загальної корпоративної підсистеми подій та вчасне інформування про важливі тригери в межах програмного засобу аналізу та прогнозування надійності ПС.

Для забезпечення максимальної гнучкості введення даних інженери також передбачили окремий модуль імпорту звітів з використанням файлів формату MS Excel. Це дає змогу завантажувати інформацію про виявлені помилки в автономному режимі, оминаючи пряме підключення до баг-трекера. Щоб оптимізувати загальну продуктивність системи під час роботи з великими статистичними масивами, було впроваджено модуль паралельного опрацювання імпортованих звітів про помилки. Він ефективно розпаралелює обчислювальні потоки, що дозволяє суттєво пришвидшити роботу всього механізму імпортування даних. Нарешті, взаємодію людини з машиною забезпечує модуль побудови веб-орієнтованого інтерфейсу. Саме через цей зручний графічний інтерфейс користувач здійснює всі операції з керування процесом імпортування звітів. Варто наголосити, що весь цей комплексний програмний засіб спроектовано та реалізовано з глибоким дотриманням усіх класичних принципів об'єктно-орієнтованого підходу, що гарантує його надійність та здатність до подальшого масштабування.

2.3. Використання засобів автоматизації фірми ІВА для діагностики неполадок

У реальних умовах сучасного промислового виробництва класичні

інструменти діагностики стану технічного обладнання часто виявляються недостатньо ефективними. Традиційні системи аварійних сповіщень, що виводяться на робочу станцію оператора, зазвичай працюють за жорстко заданими алгоритмами і здатні ідентифікувати лише заздалегідь змодельовані типові несправності. Проте вони фізично не можуть передбачити абсолютно всі теоретично ймовірні комбінації факторів, що призводять до виникнення позаштатних чи аварійних ситуацій на виробничій лінії. Згідно з аналізом актуальних літературних і технічних джерел, одним із найперспективніших рішень у цій сфері є впровадження спеціалізованих систем високошвидкісного збору та надійного зберігання даних, розробкою яких активно займається компанія ІВА. У контексті промислової автоматизації головним джерелом об'єктивної інформації виступає безпосередньо прикладна програма промислового логічного контролера. Наприклад, базова система автоматизації типового трубопрокатного цеху функціонує на базі надійних програмованих контролерів лінійки Simatic від відомої корпорації Siemens. Саме у внутрішній пам'яті цих контролерів акумулюється і безперервно оновлюється весь масив даних, необхідних для коректного управління безперервним технологічним процесом.

Впровадження виділеного сервера ІВА для безперервного збору та архівації цих контролерних параметрів відкриває перед виробничим підприємством цілу низку вагомих переваг. Передусім, інженери отримують потужний інструмент для виявлення та діагностики неявних, «плаваючих» несправностей, які неможливо зафіксувати звичайними методами. Також система забезпечує можливість глибокого ретроспективного аналізу складних каскадних збоїв і технологічних аварій. Завдяки розширеному інструментарію фахівці можуть детально досліджувати внутрішню логіку виконання надзвичайно складних алгоритмів керування, що стає критично важливим на етапі пусконаладжувальних робіт або під час впровадження нових експлуатаційних режимів. Додатковим бонусом є висока загальна зручність у використанні системи: вона дозволяє будувати наочні графіки динаміки будь-яких фізичних величин і формувати структурований архів

історичних даних, термін зберігання інформації в якому є практично необмеженим.

Традиційно процес пошуку несправності виглядає так: інженер-електронік підключається до обладнання і здійснює візуальний моніторинг виконання програми керуючого контролера безпосередньо в середовищі розробки Step7. Однак такий класичний підхід має цілий ряд суттєвих недоліків. Зокрема, інженер абсолютно позбавлений можливості переглянути хронологію аварії, що вже сталася в минулому. Крім того, у стандартному середовищі відсутнє зручне графічне відображення аналогових сигналів, неможливо одночасно переглядати та комплексно аналізувати велику кількість змінних, а сама швидкість обробки діагностичних даних є доволі низькою.

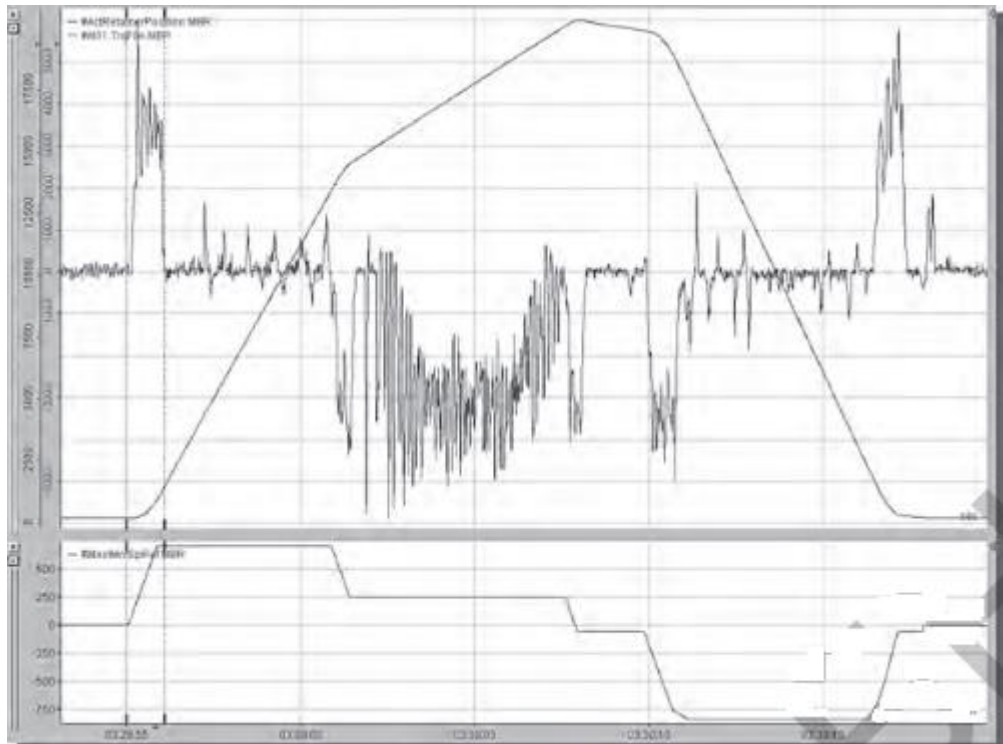


Рис. 2.4. Приклад тренду програми IBA-analyzer

Застосування передових засобів діагностики набуває особливої, критичної актуальності для таких складних ділянок, як лінія гарячої прокатки труб. Технологічний ланцюжок виробництва гарячекатаної труби на цій ділянці є строго послідовним і неперервним. Це означає, що аварійна зупинка навіть одного проміжного прокатного стану неминуче призведе до повної зупинки всієї величезної ділянки. У такій стресовій ситуації саме

максимально швидко і безпомилково діагностика першопричини несправності дозволяє радикально скоротити час вимушеного простою дорогого обладнання. Часто під час локалізації проблеми виникає гостра необхідність у паралельному аналізі графіків двох, трьох і більше фізичних величин. Алгоритм функціонування трубопрокатного комплексу є багатовимірним і складається з величезної кількості послідовних кроків та жорстких логічних умов. Для розуміння масштабу: лише одна послідовність циклу прокатки рейки ретайнера містить 17 постійних блокуючих умов і складається з 28 окремих робочих кроків. Переходи між цими кроками також є умовними і залежать від багатьох датчиків. Всю цю надзвичайно складну багаторівневу логіку роботи можна легко візуалізувати та проаналізувати на екрані комп'ютера саме завдяки детальному запису трендів у системі ІВА.

Процес діагностики несправностей за допомогою цього інструментарію можна здійснювати у двох базових режимах: онлайн (моніторинг у режимі реального часу) та офлайн (глибокий перегляд історичного архіву даних). Під час інтенсивної прокатки регулярно виникають такі специфічні несправності, які фізично неможливо виявити та усвідомити в режимі реального часу — наприклад, раптове екстрене гальмування розкатного стану. До моменту, коли на місце інциденту прибувають ремонтні бригади, все обладнання вже повністю знеструмлене та зупинене. Спираючись лише на сухі текстові повідомлення журналу на станції оператора, встановити справжню причину зупинки вдається далеко не завжди. Саме в таких складних випадках ретельний перегляд збережених графічних трендів за допомогою програмного забезпечення ІВА-analyzer (рис. 2.4) стає абсолютно незамінним етапом розслідування інциденту.

Унікальна здатність комплексу ІВА репрезентувати величезні масиви даних у вигляді зрозумілих графічних закономірностей дає змогу інженерам миттєво діагностувати різноманітні механічні дефекти обладнання. На графіках чітко видно мікропідклинювання важких рухомих частин, нетипові удари в механічних з'єднаннях чи надмірно утруднений хід виконавчих механізмів. Оскільки система здатна зчитувати інформацію безпосередньо із

швидкісної промислової шини PROFIBUS, запис трендів відбувається з неймовірною дискретністю — до 10 мілісекунд. Така надвисока частота опитування робить можливим гарантоване виявлення навіть найменших, мілісекундних зникнень сигналів від індуктивних датчиків або короточасних небезпечних кидків струмового навантаження на головному приводі.

Не варто забувати і про те, що дуже часто головною причиною випуску бракованої невідповідної продукції або виникнення серйозних збоїв у роботі механізмів стає саме горезвісний людський фактор. Це може бути банальна відсутність належного контролю за роботою лінії, помилкове введення та підтвердження некоректних технологічних параметрів, або ж порушення оператором правильної послідовності дій під час ручного управління механізмами. Якщо на підприємстві відсутні детальні архівні записи, що дозволяють крок за кроком оцінити правильність виконання персоналом вимог посадових та технологічних інструкцій, то довести провину або виявити точну причину збою буває практично неможливо. Однак, якщо всі без винятку дії оператора автоматично фіксуються та дублюються на ІВА-сервері, то під час розслідування технологи можуть легко відтворити події: які саме дії, в якій послідовності та в які точні мілісекунди часу виконувала людина.

Система неупереджено фіксує кожне натискання кнопок, будь-які спроби зміни робочих режимів як на головному, так і на місцевих пультах управління, введення нових уставних параметрів на станції оператора, а також точне просторове положення всіх механізмів і загальний хід технологічного процесу саме в момент втручання технологічного персоналу. Завдяки такому всебічному та глибокому аналізу стає можливим безпомилкове виявлення несправностей будь-якої природи: від механічної та гідравлічної до складної електричної частини обладнання і систем промислової автоматики. Підсумовуючи, можна впевнено стверджувати, що впровадження передових засобів автоматизації фірми ІВА для комплексної діагностики несправностей устаткування трубопрокатного цеху робить

процедуру пошуку дефектів на порядок швидшою, зручнішою та математично точною, тим самим кардинально скорочуючи час простоїв обладнання і мінімізуючи виробничий брак.

2.4. Досвід застосування Excel та MathCad для встановлення надійності телекомунікаційних пристроїв

Безперервний та стрімкий розвиток сучасних мобільних терміналів, смартфонів і високопродуктивної комп'ютерної техніки неминуче диктує нові жорсткі правила для світового ринку зв'язку. Експоненційне зростання очікувань кінцевих споживачів щодо пропускної здатності каналів, а також постійне підвищення стандартів якості та швидкості надання інтегрованих інфокомунікаційних послуг стають головними рушійними силами технічного прогресу в цій галузі. Саме ці об'єктивні фактори виступають фундаментальною причиною для проведення масштабної та перманентної модернізації існуючих телекомунікаційних систем, розширення транспортних мереж та швидкого впровадження інноваційних технологій передачі даних.

У цьому контексті критично важливим інженерним завданням стає не просто фізичне оновлення апаратної бази, але й глибока оптимізація архітектурних структур мереж на етапі їх проектування. Фахівці змушені безперервно аналізувати, розраховувати та покращувати ключові експлуатаційні показники обладнання, а також ретельно досліджувати всі ймовірні чинники (як зовнішні, так і внутрішні), що здатні здійснювати деструктивний вплив на загальну стабільність роботи телекомунікаційних вузлів. Для вирішення таких складних аналітичних та ймовірнісних задач на практиці активно залучаються потужні інструментальні засоби та математичні середовища, зокрема табличні процесори рівня Microsoft Excel та спеціалізовані математичні пакети на кшталт MathCad. Їх використання дозволяє автоматизувати рутинні обчислення, будувати адекватні математичні моделі процесів відмов та з високою точністю встановлювати реальні межі надійності сучасних телекомунікаційних пристроїв.

3. АВТОМАТИЗАЦІЯ ДІЯЛЬНОСТІ МАЛИХ ПІДПРИЄМСТВ НА ПРИКЛАДІ ФІРМИ SIGMA

3.1. Перспективні напрями автоматизації роботи малого бізнесу

Дослідження процесів автоматизації доцільно розглянути на реальному прикладі вітчизняного підприємства. Об'єктом вивчення обрано компанію Sigma, яка успішно функціонує на ринку вже понад десять років і є впізнаваним українським брендом у сфері інформаційних технологій, сучасної телефонії та засобів радіозв'язку. Стратегічною місією цього підприємства є не лише створення надійної та високоякісної апаратури, але й активна популяризація національного ІТ-продукту серед вітчизняних та закордонних споживачів. Керівництво компанії спрямовує свої зусилля на забезпечення сталого розвитку бізнесу та загальне зміцнення вітчизняного технологічного потенціалу. Щоб максимально точно відповідати запитам цільової аудиторії та своєчасно реагувати на глобальні світові тренди, на підприємстві активно функціонує спеціалізований підрозділ досліджень і розробок (R&D). Його фахівці безперервно тестують технологічні новинки, прагнучи досягти і ідеального балансу між ціною та якістю для задоволення потреб амбасадорів бренду Sigma mobile. За своїм масштабом фірма класифікується як мале підприємство, оскільки її постійний штат налічує близько 30 висококваліфікованих співробітників. Організаційна структура компанії є доволі розгалуженою і включає відділ стратегічного управління, службу логістики, сервісний центр, згаданий раніше R&D підрозділ, а також комерційну групу менеджерів, які відповідають за безпосередній збут готової продукції.

Вирішуючи комплексне завдання щодо автоматизації операційної діяльності цього малого підприємства, було здійснено глибокий аналіз фахових літературних джерел та експертних думок у цій галузі. Сучасні дослідники та аналітики сходяться на думці, що для побудови дійсно ефективної системи управління господарськими та фінансовими потоками підприємства, критично необхідним є впровадження сучасного інформаційно-програмного забезпечення [18]. Така комплексна

діджиталізація має охоплювати всі без винятку структурні елементи бізнесу: від забезпечення стабільної роботи філій та профільних підрозділів до організації надійної системи внутрішньої комунікації та службових розсилок.

У процесі детального вивчення внутрішніх бізнес-процесів фірми Sigma було чітко ідентифіковано ті ключові відділи та операційні ланки, які першочергово потребують структурної реорганізації та модернізації шляхом глибокої інтеграції новітніх інформаційних технологій. Стало очевидним, що ручне керування окремими процесами гальмує загальний розвиток компанії та знижує ефективність взаємодії між відділами.

Спираючись на систематизовані дані з передових наукових джерел, а також акумулювавши практичний досвід щодо впровадження систем автоматизації в секторі малого бізнесу [16, 23, 50], було сформовано концептуальне бачення цифрової трансформації компанії. Використовуючи базові принципи побудови інформаційних систем та практичні навички проектування програмних продуктів, нами було розроблено концепцію та запропоновано комплекс спеціалізованих програмних рішень. Ці рішення спрямовані на наскрізну автоматизацію робочих процесів малого підприємства на різних рівнях його організаційної ієрархії. Наочну структурну схему запропонованих рішень для оптимізації діяльності фірми Sigma детально проілюстровано на рисунку Рис. 3.1.

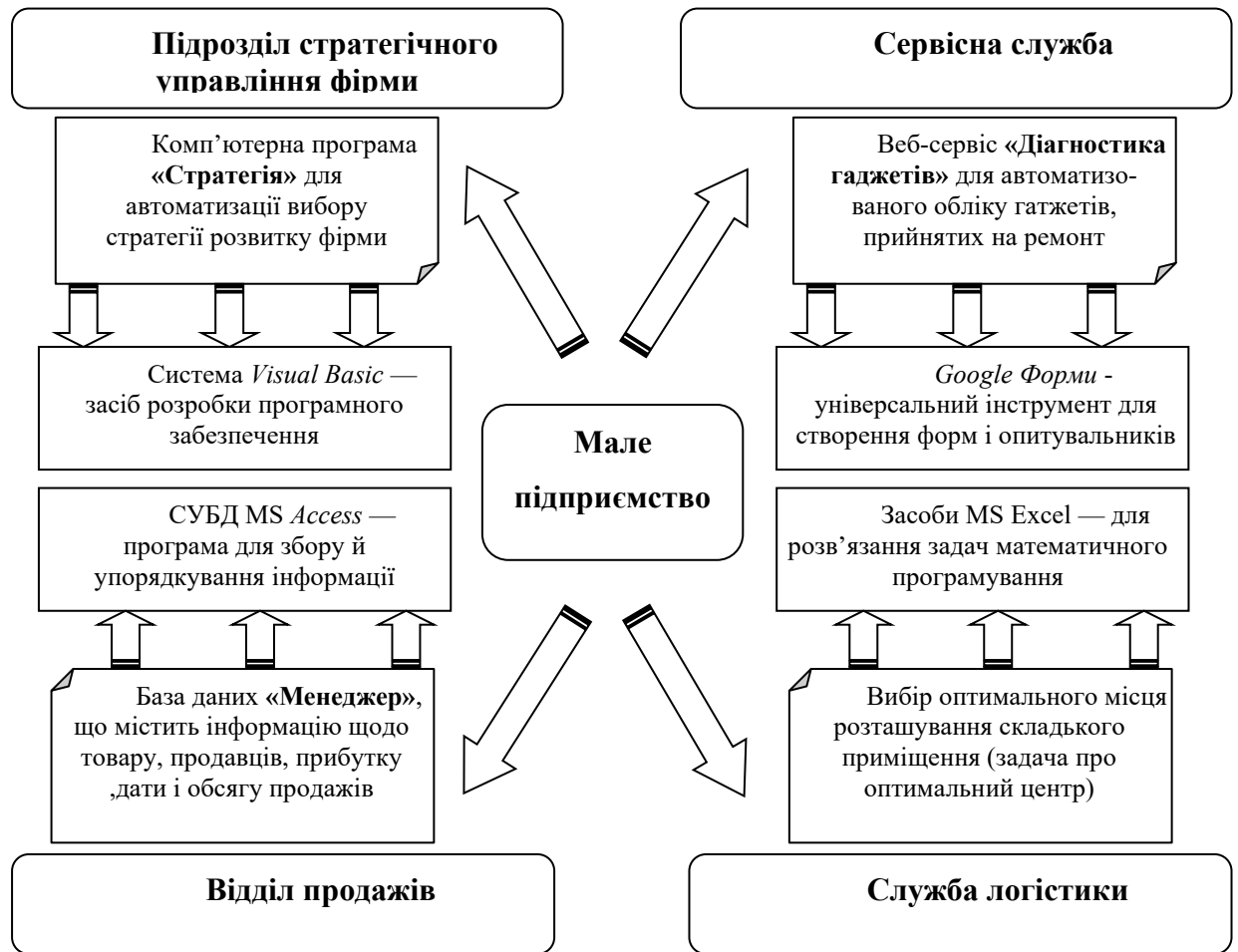


Рис. 3.1. Автоматизація роботи малого бізнесу на різних рівнях організації (на прикладі роботи фірми Sigma)

Зокрема, для ефективної автоматизації роботи підрозділу стратегічного управління фірми було розроблено та запропоновано до впровадження спеціалізовану комп'ютерну програму під назвою «Стратегія». Головне призначення цього програмного продукту полягає в автоматизації складного аналітичного процесу вибору оптимального вектора та сценаріїв розвитку підприємства. Програму створено з використанням середовища розробки Visual Basic. Цей інструмент є надзвичайно потужним та гнучким рішенням для компонентного підходу до проектування застосунків, що повноцінно функціонують під управлінням операційних систем сімейства Microsoft Windows. Варто наголосити, що алгоритмічною та математичною основою запропонованої програми виступає визнаний у сучасному менеджменті метод SPACE-аналізу (Strategic Position and Action Evaluation). Практичне застосування цієї програми дозволяє керівництву швидко та обґрунтовано

розв'язувати комплексні завдання, пов'язані з автоматизацією стратегічного управління фірмою в умовах висококонкурентного та динамічного ринку.

Наступним важливим кроком стала диджиталізація та модернізація технічної підтримки клієнтів. З метою повної автоматизації рутинної роботи сервісної служби було спроектовано та успішно розгорнуто зручний Веб-сервіс «Діагностика гаджетів». Завдяки цьому мережевому ресурсу інженери та приймальники здійснюють строгий і прозорий облік усіх пристроїв, що надходять на гарантійний чи післягарантійний ремонт. У базі даних веб-сервісу обов'язково фіксується точна назва та модель кожного гаджета, поточна версія його програмної прошивки, а також дата проведення вхідного тестування. Крім базової реєстрації пристроїв, система надає можливість автоматичної генерації готових звітів за затвердженими на підприємстві стандартними формами. Більше того, сервіс дозволяє формувати глибокі статистичні звіти щодо найпоширеніших апаратних дефектів і типових програмних несправностей обладнання, що є безцінною аналітичною базою для R&D підрозділу.

Для потреб комерційного відділу було розроблено повноцінну реляційну базу даних під робочою назвою «Менеджер». Її фізична реалізація виконана на базі популярної системи управління базами даних MS Access. Цей програмний інструмент спрямований на всебічну автоматизацію повсякденної роботи відділу продажів. База акумулює в собі вичерпну інформацію щодо наявного асортименту товарів, персональних даних та KPI продавців, а також точних дат та обсягів реалізованої продукції. Впровадження БД «Менеджер» дозволяє в автоматичному режимі розраховувати чистий прибуток компанії за будь-який обраний звітний період, об'єктивно оцінювати динаміку та рівень продажів конкретної моделі гаджета, а також аналізувати персональну ефективність кожного окремого менеджера. Додатково система допомагає виявляти поточні тренди популярності товарів серед покупців, оперативно контролювати фактичні залишки продукції на складах та миттєво генерувати зведені звіти з товарообігу.

Вирішення логістичних завдань компанії також зазнало цифрової трансформації. Для суттєвої оптимізації транспортних перевезень, маршрутизації та зниження витрат на доставку продукції було запропоновано раціонально використовувати вбудовану аналітичну надбудову «Пошук рішень» (Solver), що входить до стандартного пакету табличного процесора MS Excel. Цей математичний інструмент дозволяє швидко та ефективно автоматизувати процес розв'язання класичної транспортної задачі, знаходячи оптимальні логістичні маршрути та мінімізуючи транспортні витрати малого підприємства без необхідності закупівлі дорогого вузькоспеціалізованого софту.

Підсумовуючи проведену роботу, можна впевнено стверджувати, що в ході даного дослідження було представлено цілий комплекс відносно простих, але вкрай дієвих програмних рішень. Їхня головна перевага полягає в тому, що вони практично не потребують додаткових значних капіталовкладень на придбання ліцензій на важкі ERP-системи чи розгортання складної серверної інфраструктури. Незважаючи на свою економічну доступність, запропоновані інструменти довели свою високу ефективність у вирішенні глобального завдання з комплексної автоматизації операційної, сервісної, комерційної та стратегічної діяльності сучасної фірми.

3.2. Автоматизація роботи підрозділу стратегічного управління

Ефективний та стабільний розвиток будь-якого сучасного підприємства безпосередньо залежить від впровадження новітніх підходів до стратегічного управління. Володіючи об'єктивною та вичерпною інформацією щодо власних конкурентних позицій на ринку, компанія здобуває реальну можливість чітко визначати пріоритетні напрями вдосконалення наявних переваг. Це, у свою чергу, дозволяє розробити дієву стратегію, яка буде максимально адаптованою та адекватною до суворих реалій сучасних динамічних ринкових процесів [16].

З метою практичного розв'язання складного завдання щодо

автоматизації процесів стратегічного управління у фірмі Sigma, нами було розроблено та запропоновано спеціалізовану комп'ютерну програму. Фундаментальною логічною основою цього програмного продукту виступає визнаний у світовій практиці метод SPACE-аналізу, який гармонійно поєднано з методом експертних оцінок. З практичної точки зору, застосування алгоритму SPACE-аналізу вимагає послідовного виконання кількох ключових етапів: збору первинних експертних оцінок, наступного математичного розрахунку усереднених значень для кожної групи факторів, графічної побудови багатокутника (полігону) оцінок на координатній площині, а також фінального обчислення та побудови вектора, який вказує на оптимальну рекомендовану стратегію.

Варто відзначити, що до складу робочої експертної групи, яка бере участь в оцінюванні, обов'язково входить керівник відділу стратегічного планування, його безпосередній заступник та провідний фінансовий аналітик компанії. Під час детального вивчення класичного процесу вибору вектора розвитку підприємства було виявлено, що на етапі проведення розрахунків фахівці стикаються з певними рутинними складнощами. Навіть за умови активного використання головою стратегічного відділу стандартних табличних процесорів рівня MS Office Excel, значна частина аналітичної роботи не піддавалася базовій автоматизації і вимагала тривалого ручного втручання.

Відповідно до класичної процедури, після внесення до робочої таблиці всіх необхідних вихідних даних, які комплексно характеризують зовнішнє середовище організації та визначають її стратегічне позиціонування на ринку, експерти виставляють бали за шкалою від -6 до 6. На основі цих зібраних даних здійснюється математичний розрахунок середніх арифметичних оцінок за кожною виділеною групою факторів, що наочно зведено у таблиці нижче.

Таблиця 3.1

Результати експертної оцінки факторів методом SPACE-аналізу

	Фактори	Експерт 1	Експерт 2	Експерт 3	Середні оцінки
ES	зміни в технології	1	2	1	
	варіативність попиту	-1	2	0	
	ціновий діапазон конкурентних продуктів	5	2	6	2,00
CA	ринкова частка	-4	-2	-5	
	якість продукту	4	2	3	
	лояльність покупця	2	3	1	0,44
IS	потенціал росту	2	4	1	
	потенціал прибутку	1	-1	0	
	фінансова стабільність	2	0	4	1,44
FS	використання кредитів	4	3	5	
	ліквідність	-2	3	2	
	ризик бізнесу	5	5	4	3,22

На наступному аналітичному етапі відбувається безпосереднє визначення найбільш прийнятної стратегії для фірми Sigma із застосуванням математичного апарату SPACE-аналізу. Розрахунок підсумкового вектора рекомендованої стратегії здійснюється за допомогою наведених нижче формул, які дозволяють математично точно описати координати ключових точок у двовимірній декартовій площині цієї стратегічної моделі. Оцінки координат вектора визначаються за такими класичними залежностями:

$$X = IS - CA, \quad (3.1)$$

$$Y = FS - ES, \quad (3.2)$$

Підставивши отримані експертні значення, маємо такий результат практичних розрахунків: по осі абсцис значення становить $X = 1,44 - 0,44 = 1,00$, а по осі ординат $Y = 3,22 - 2,0 = 1,22$.

Виходячи з отриманих результатів, координати точок, що формують та визначають просторовий вектор рекомендованої стратегії розвитку фірми за методом SPACE, розподіляються наступним чином. Точка початку вектора (точка А) завжди незмінно розташовується в центрі початку координат. Своєю чергою, кінець вектора (точка В) фіксується в точці з розрахованими координатами (1,00; 1,22). Це математично означає, що результуючий вектор стратегії належить прямій, яка описується рівнянням $Y = 1,22x$. Графічне

відображення сформованого полігону матриці SPACE, а також наочна побудова фінального вектора рекомендованої стратегії детально представлені на рисунку нижче.

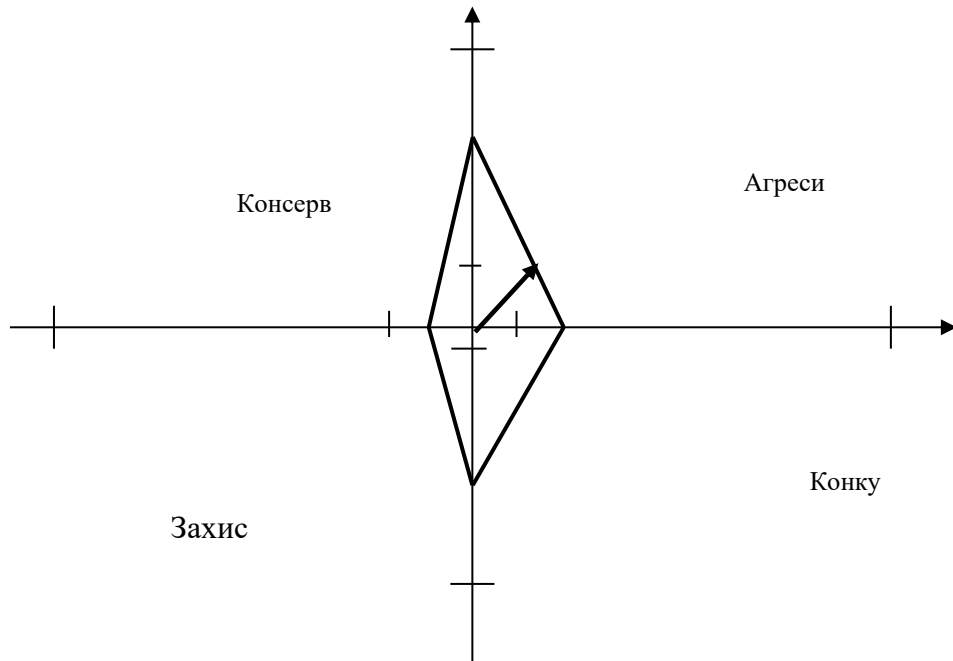


Рис. 3.2 – Полігон матриці та побудова вектора рекомендованої стратегії за методикою SPACE-аналізу

Провівши глибоке ознайомлення з усіма аспектами процесу вибору стратегії розвитку фірми та усвідомивши обсяг рутинних обчислень, було зроблено логічний висновок про гостру необхідність спрощення цієї процедури. Саме тому було запропоновано та успішно реалізовано ідею повної автоматизації даного процесу шляхом створення власної комп'ютерної програми. Цей програмний продукт, що отримав робочу назву «Стратегія», було спроектовано та написано з використанням середовища розробки Visual Basic. Даний вибір зумовлений тим, що ця система є максимально простим, інтуїтивно зрозумілим та доступним інструментальним засобом для швидкого створення прикладного програмного забезпечення.

Архітектурно розроблена комп'ютерна програма складається з чотирьох основних логічних блоків. Перший блок відповідає за безпосереднє введення оцінок факторів членами експертної комісії. Загалом анкета містить 12 запитань, за допомогою яких експертно оцінюються по три фактори з кожної

категорії, що всебічно характеризують як макросередовище організації, так і її внутрішнє стратегічне позиціювання. До цих категорій належать: фактори, що визначають загальну стабільність середовища (ES); фактори, що формують конкурентні переваги підприємства (CA); показники, що характеризують сильні сторони самого виробництва (IS); а також критерії, що визначають фінансову силу компанії (FS). Другий функціональний блок програми забезпечує автоматичний математичний розрахунок середніх оцінок за всіма чотирма згаданими групами факторів. Третій блок відповідає за геометричне знаходження та побудову вектора рекомендованої стратегії фірми згідно з класичною методикою SPACE. Нарешті, четвертий блок здійснює фінальний аналіз та генерує висновок щодо вибору оптимальної стратегії. Покроковий алгоритм роботи цього програмного комплексу наочно проілюстровано у вигляді блок-схеми на рисунку 3.3.

Як можна побачити з ілюстративного матеріалу, запропонована програма вирізняється максимально простим, лаконічним та інтуїтивно зрозумілим користувацьким інтерфейсом. Головне робоче вікно містить три основні керуючі кнопки та спеціальну текстову область для оперативного виведення рекомендацій на екран монітора (рис. 3.4). У процесі експлуатації програми сформована група експертів у діалоговому режимі по чергово оцінює кожен із ключових факторів, на основі яких традиційно здійснюється комплексний аудит поточного стану малого підприємства (рис. 3.5). Відразу після того, як експерти успішно завершать введення всіх необхідних оцінок до бази, програмний алгоритм миттєво обробляє масив даних і виводить на екран підсумкове інформаційне повідомлення.

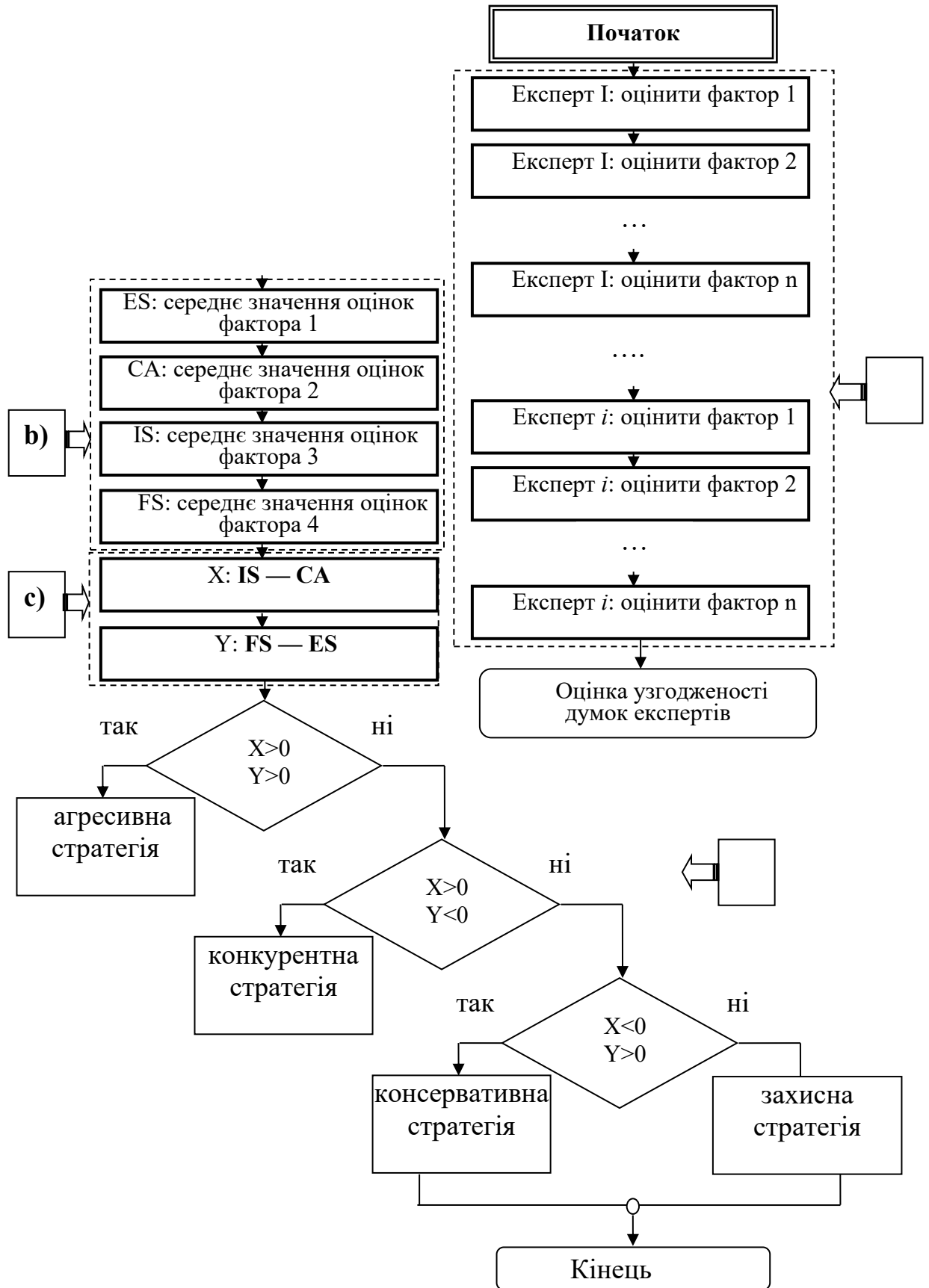


Рис. 3.3 – Алгоритм комп'ютерної програми вибору стратегії розвитку фірми

У цьому вікні чітко зазначається, яку саме глобальну стратегію поведінки на ринку рекомендовано обрати фірмі, зважаючи на її поточний фінансовий та конкурентний стан (рис. 3.6).

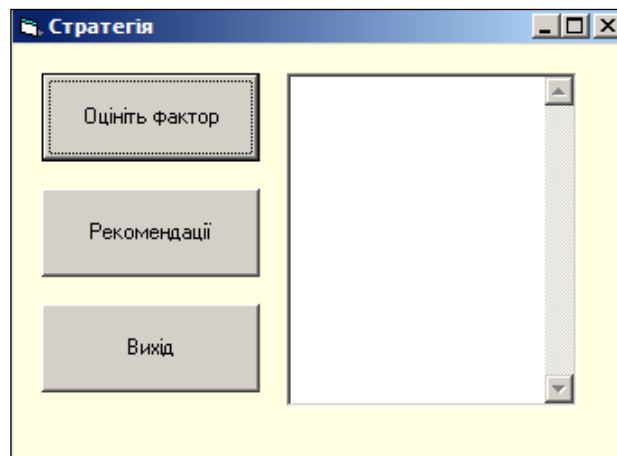


Рис. 3.4 – Головне вікно комп'ютерної програми «Стратегія»

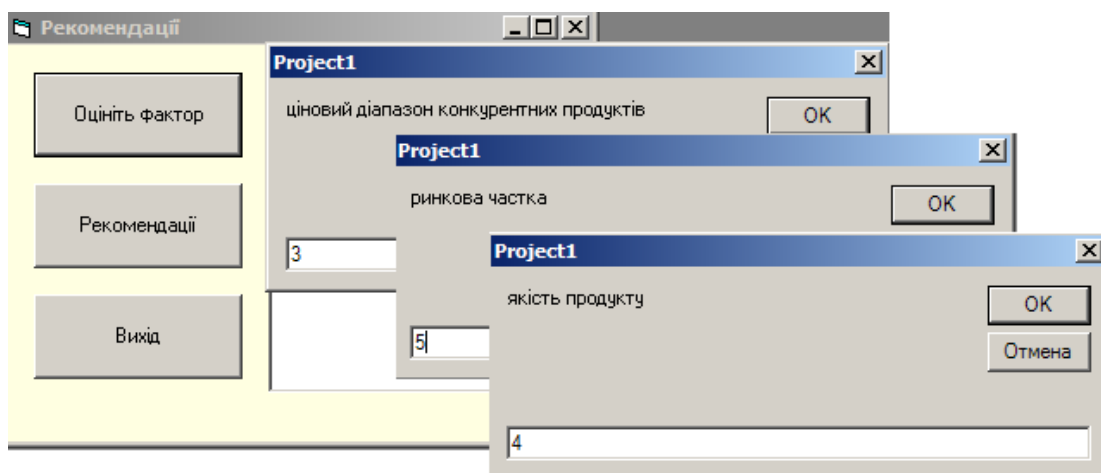


Рис. 3.5 – Визначення експертних оцінок

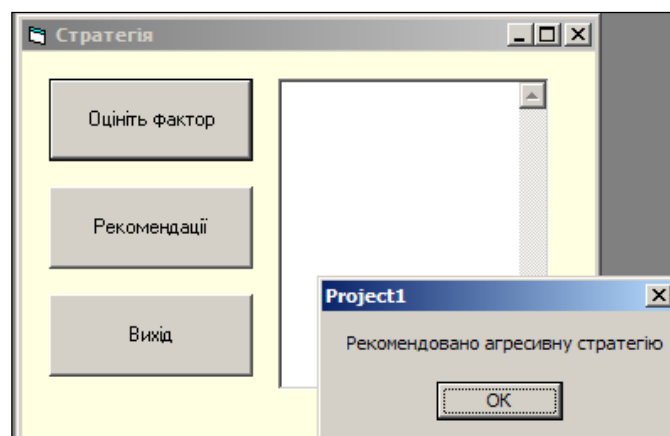


Рис. 3.6 – Отримання повідомлення про рекомендовану стратегію

Подальша взаємодія з інтерфейсом передбачає натискання керуючої кнопки «Рекомендація», що призводить до появи нового діалогового вікна, в яке користувачеві необхідно вручну ввести назву згенерованої системою стратегії. У програмі реалізовано базовий захист від помилок: якщо назву буде введено некоректно, система миттєво видасть попереджувальне повідомлення про помилку. Якщо ж назву введено правильно, у головному текстовому полі програми розгортається детальна довідкова інформація, де фахівці можуть ознайомитися з головними напрямками тактичних дій, які повністю відповідають обраній рекомендованій стратегії (рис. 3.7). За результатами практичного застосування цієї комп'ютерної програми з реальними даними компанії, було отримано однозначну машинну рекомендацію: на поточному етапі свого життєвого циклу фірмі необхідно дотримуватися агресивної стратегії розвитку. Спираючись на цей об'єктивний висновок, плановий відділ підприємства здійснює подальший глибокий аналіз та розробляє конкретний план заходів для імплементації цієї стратегії у бізнес-процеси.

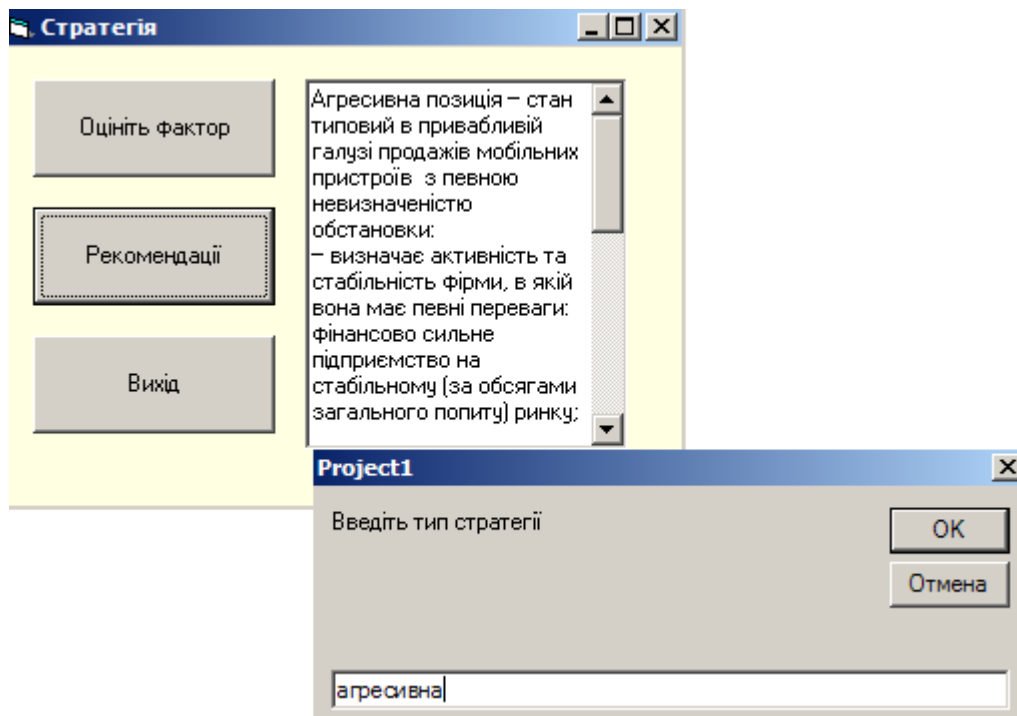


Рис. 3.7 – Відомості про рекомендовану стратегію

Згідно з теорією стратегічного менеджменту, агресивна позиція є абсолютно типовим і бажаним станом для компаній, що оперують у

фінансово привабливій галузі, такій як ринок продажів сучасних мобільних пристроїв, навіть за умов наявності певної зовнішньої невизначеності обстановки. Ця стратегія чітко ідентифікує високу ділову активність та фінансову стабільність фірми, підкреслюючи, що вона має вагомі конкурентні переваги та позиціонує себе як фінансово міцне підприємство на стабільному (з точки зору обсягів загального попиту) ринку. Разом з тим, найкритичнішим фактором ризику для такої позиції залишається раптова поява нових потужних гравців-конкурентів. Однак поточна сильна позиція дозволяє компанії надзвичайно активно та гнучко реагувати на будь-які зміни ринкової кон'юнктури, планомірно збільшувати свою частку на ринку та невинно підвищувати якість і загальну конкурентоспроможність власної продукції. У підсумку, фірма отримує міцні конкурентні переваги, які вона здатна не лише зберегти в довгостроковій перспективі, але й суттєво примножити, грамотно використовуючи свій потужний фінансовий потенціал. Оскільки зовнішні загрози в такому стані є не досить значними, головним вектором зусиль менеджменту має стати тотальна концентрація всіх наявних ресурсів на забезпеченні та розширенні стратегічних інтересів компанії.

3.3 Удосконалення діяльності служби діагностики гаджетів

У процесі комплексного дослідження операційної діяльності фірми Sigma-mobile та пошуку шляхів її оптимізації, нами було розроблено та запропоновано авторське програмне рішення. Це рішення повністю ґрунтується на сучасних хмарних технологіях та можливостях інтерактивних веб-сервісів. Ініціатором цього напрямку модернізації виступив головний інженер компанії, який поставив перед нашою командою чітке завдання: розробити зручний, надійний та масштабований інструмент на базі доступних веб-сервісів для оперативного збору, систематизації та подальшого глибокого аналізу даних, отриманих під час діагностики мобільного обладнання.

На етапі підбору найбільш доцільного та економічно виправданого

ресурсу, який би дозволив технічно реалізувати поставлене завдання, ми провели порівняльний аналіз кількох платформ. У підсумку наш вибір зупинився на одному з найпопулярніших хмарних сервісів від корпорації Google — Google Формх (Google Forms). Причина такого вибору була цілком аргументованою: компанія Google успішно інтегрувала у єдиний екосистемний пакет усі необхідні інструменти для продуктивної та синхронізованої роботи. Ця платформа гарантує безшовну роботу на будь-яких пристроях — від стаціонарних комп'ютерів до смартфонів і планшетів. Тому рішення використати цю універсальну платформу було однозначним, адже вона автоматично забезпечує міцну «зв'язку» з усіма супутніми робочими ресурсами компанії: електронними таблицями для розрахунків, поштовими клієнтами для розсилок, хмарними сховищами тощо. Не менш важливим фактором стало те, що Google надає легальну можливість використовувати переважну більшість свого функціоналу абсолютно безкоштовно, що є критично важливою перевагою для малого бізнесу на етапі оптимізації витрат та розгортання нових ІТ-рішень (рис. 3.8).

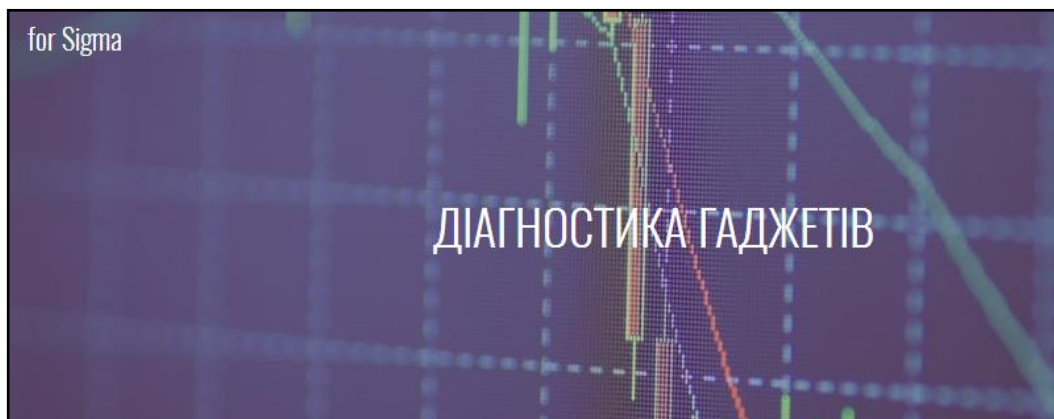


Рис. 3.8 – Діагностика гаджетів

Процес розгортання та використання системи є максимально спрощеним. Для початку роботи інженерам сервісного центру необхідно пройти стандартну процедуру реєстрації нового акаунту або ж просто виконати ідентифікацію (авторизацію) в системі, якщо користувач уже має зареєстрований корпоративний профіль Google (рис. 3.9).

Рис. 3.9 – Реєстрація на сайті

Для повноцінного розуміння архітектури запропонованого рішення варто пояснити, що саме являє собою даний онлайн-конструктор. Процес створення складних діагностичних форм опитування в цьому середовищі є настільки ж інтуїтивним і легким, як і набір звичайного текстового документа. Під час практичної реалізації цього проекту перед нами постало важливе методологічне питання: в якому саме форматі найкраще представити технічні запитання респонденту (інженеру-тестувальнику), та який тип користувацького інтерфейсу буде найбільш ергономічним для швидкої роботи з цим веб-сервісом (рис. 3.10).

Рис. 3.10 – Головне меню

Інструментарій Google Форм надає розробникам надзвичайно широкий спектр можливостей. Зокрема, платформа дозволяє конструювати високопрофесійні опитувальники з розгалуженою логікою. Це дає змогу сервісній службі отримувати високоточні статистичні дані, що, у свою чергу, є головною запорукою формування глибокої довіри з боку клієнтів компанії. Сервіс дозволяє неймовірно швидко та зручно додавати різноманітні варіанти відповідей, вибирати специфічні типи запитань (наприклад, шкали, списки, матриці) та легко змінювати їхню логічну послідовність. Більше того, для кращого візуального сприйняття інструкцій можна безпосередньо у форму інтегрувати фотографії дефектів, навчальні відеоматеріали та складні логічні схеми переходів, що значно підвищує залученість та уважність персоналу під час діагностики. Усі зібрані вихідні дані миттєво синхронізуються та доступні для глибокого аналізу в Google Таблицях або ж можуть бути експортовані до інших спеціалізованих програм. Крім того, керівник сервісного центру має змогу здійснювати онлайн-моніторинг та переглядати відповіді інженерів у режимі реального часу.

Рис. 3.11 – Можливості веб-сервісу

Під час поглибленого налаштування нашого діагностичного сервісу ми активували низку критично важливих параметрів безпеки та верифікації. Було увімкнено функцію «Збирати адреси електронної пошти», що дозволяє точно ідентифікувати працівника, який проводив тест. Також, для

збереження історії перевірок, ми налаштували опцію відправки респондентам копій їхніх звітів після завершення опитування. Щоб уникнути дублювання даних та можливих маніпуляцій зі звітами, ми встановили жорстке правило: респондентам обов'язково потрібно виконати вхід у корпоративний акаунт Google, і вони мають право відправляти заповнену форму не більше одного разу для конкретного серійного номера пристрою. Ще однією надзвичайно корисною функцією стала можливість для користувача, після повного проходження алгоритму діагностики, переглядати зведену статистику та автоматично згенеровані кругові діаграми типових несправностей.

Потужності серверів Google Форм дозволяють нашому сервісному центру опрацьовувати колосальні обсяги технічної інформації без ризику зависання системи. Наразі архітектура рішення здатна безперебійно обробляти такі ж масиви даних, що й класичні Google Таблиці, а це означає підтримку до двох мільйонів заповнених клітинок з даними в межах одного документа, що є більш ніж достатнім навіть для великого підприємства.

Оскільки в базових налаштуваннях веб-сервісу ми жорстко зафіксували опцію обов'язкового збору електронних адрес, процес діагностики завжди розпочинається з авторизації пошти інженера. Одразу після цього відкривається вступний блок, що містить кілька ідентифікаційних запитань: працівник повинен точно вказати комерційну назву та модель гаджета, поточну версію встановленої програмної прошивки та фактичну дату проведення тестування. Лише виконавши ці базові формальності, фахівець отримує доступ до основної, найважливішої частини технічного тесту. Цей блок складається з дванадцяти спеціалізованих уточнюючих запитань, які відображають найчастіші та найкритичніші проблеми під час діагностики прошивок мобільних телефонів Sigma. Оскільки деякі тестові процедури мають доволі специфічні інженерні назви, для зручності персоналу біля кожного пункту передбачено розширену текстову підказку. У ній детально покроково розписано, які саме фізичні чи програмні маніпуляції з телефоном треба виконати, щоб отримати об'єктивний результат перевірки.

Основними пунктами чек-листа діагностики є наступні перевірки. По-

перше, тестується регістро-незалежний пошук у кирилиці: інженер перевіряє коректність роботи алгоритмів пошуку контактів незалежно від того, вводиться запит великими чи малими літерами, з особливою увагою до специфічних українських літер «і, є, ї». По-друге, перевіряється алгоритм визначення контакту при вхідному дзвінку: чи коректно система ідентифікує та висвічує на екрані ім'я абонента, збереженого в телефонній книзі в локальному форматі (наприклад, 8068---*), якщо дзвінок надходить з українського номера. Далі обов'язково контролюється наявність усіх без винятку літер українського, англійського та російського алфавітів на віртуальній клавіатурі під час набору SMS-повідомлень та створення нових контактів.

Окремий великий блок присвячено аудіосистемі. Інженер перевіряє гучність та чистоту звучання як розмовного (внутрішнього), так і поліфонічного (зовнішнього) динаміків, а також загальну якість передачі голосу через мікрофон до співрозмовника. Ретельно тестується сам звук і плавність його регулювання в різних режимах (під час телефонної розмови, роботи FM-радіо та MP3-плеєра) у всьому діапазоні від мінімуму до максимуму, при цьому інженер має зафіксувати явну акустичну різницю між найгучнішим і найтихішим рівнями. Наступним критичним тестом є перевірка роботи радіомодуля з двома SIM-картами. Тестувальник зобов'язаний встановити в слоти дві активні SIM-картки (бажано від різних мобільних операторів), здійснити успішний тестовий дзвінок з кожної з них, а також переконатися у стабільності роботи апарата в режимі очікування мережі.

Крім того, діагностика охоплює перевірку базового органайзера: коректність відображення днів і дат у календарі, можливість успішно призначити нове завдання чи запланувати зустріч. Окремим пунктом стоїть перевірка можливості здійснення екстрених викликів на номери 101, 102 та 103 без наявності SIM-карти. Також тестується логіка сортування контактів у телефонній книзі при використанні як української, так і російської розкладок інтерфейсу. Важливою функцією є перевірка механізму резервного

копіювання (Backup сору), зокрема коректність масового перенесення контактів з пам'яті SIM-карти у внутрішню пам'ять телефону. Обов'язково перевіряється працездатність функції швидкого набору номерів з цифрової клавіатури. Завершується чек-лист перевіркою мультимедійних налаштувань: інженер має успішно встановити власну мелодію з зовнішньої SD-картки на загальний дзвінок або сигнал повідомлення, а також спробувати прив'язати унікальну мелодію та фотографію до окремо створеного контакту (якщо така функція підтримується апаратною платформою конкретної моделі телефону).

Даючи відповідь на кожне з цих запитань, респондент має обрати один із трьох суворо регламентованих варіантів. Варіант «Yes (done)» обирається, якщо під час практичної перевірки функція спрацювала коректно і був отриманий успішний результат. Варіант «Ні (відсутність даної функції)» фіксується у випадку, якщо програмно чи апаратно неможливо виконати даний тест на цій моделі телефону. Також передбачено поле «Інше», де інженер може вручну вписати розгорнутий коментар, зауваження чи вказати на важливу нестандартну деталь поведінки пристрою. Система налаштована таким чином, що функція захисту від «дурня» не дозволить відправити звіт на сервер, якщо інженер пропустив хоча б одне запитання — на екрані з'явиться червоне попередження «Це обов'язкове питання».

Після базового функціонального тесту йде наступний логічний блок питань, який оперує іншими статусами відповідей: «Yes (done)» (відсутність будь-яких зауважень щодо роботи модуля), «Ні (недолік)» (наявність програмного багу чи апаратної поломки) та «Ні (відсутність даної функції)». Тестувальник фізично перевіряє гаджет крок за кроком і синхронно вказує отримані результати у відповідних рядках інтерактивної таблиці.

	Yes (done)	Ні (недолік)	Ні (відсутність даної функції)
Перевірити ліхтарик:	☐	☐	☐
Перевірити роботу камери (зробити фото):	☐	☐	☐
Перевірити будильник (працює чи повтор):	☐	☐	☐
Перевірити радіо:	☐	☐	☐
Перевірити функцію запису дзвінків:	☐	☐	☐

Рис. 3.12 – Режим «питання-відповідь»

Повністю завершивши процедуру тестування та натиснувши кнопку відправки, користувач (а також керівництво сервісного центру) миттєво отримує акуратно оформлений, готовий до друку або пересилання звіт стандартної форми.

Google Forms

Благодарим за заполнение формы [Feature phones!](#)

Ваш ответ:

Feature phones

Actually, the verification algorithm with subsequent analysis of the data

Электронный адрес *

Klop95@gmail.com

Алгоритм проверки:

Вкажіть назву гаджету: *

Наприклад: "смарт. PQ-31"

Ю-67

Рис. 3.13 – Частина звіту за результатами діагностики

Величезною додатковою перевагою впровадження цього хмарного веб-сервісу є те, що він в автоматичному режимі агрегує всі надіслані звіти і дає можливість генерувати глобальний статистичний звіт про найпоширеніші

дефекти і систематичні несправності певних партій чи моделей гаджетів.

	E	F	G	H	I	J	K	L
1	1	1 Регістро-незаг	2 Визначення кс	3 Наявність всіх	4 Гучність динам	5 Звук і регулює	6 Дві SIM карти	7 Календар [1 B
2		Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)
3		Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)
4		Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)
5		Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)	Yes (done)
6								
7								

Рис. 3.14 – Частина вікна статистичного звіту

З огляду на все вищевикладене, можна зробити обґрунтований висновок: практичне застосування розробленого інтерактивного веб-сервісу «Діагностика гаджетів» кардинально пришвидшило обробку інформації, мінімізувало ризик людської помилки при заповненні паперових актів та загалом суттєво сприяло оптимізації та підвищенню якості роботи служби технічної діагностики фірми Sigma.

3.3. Застосування задач оптимізації в діяльності фірми

Вирішення стратегічного завдання щодо пошуку ідеального місця розташування для нового оптового складу продукції фірми було запропоновано розглядати крізь призму класичної математичної моделі — задачі про знаходження оптимального центру обслуговування (або центру тяжіння вантажопотоків). З базового курсу економетрії та логістики відомо, що для коректного розв'язання такого класу оптимізаційних задач критично необхідним є збір двох масивів достовірних вхідних даних: інформації про щільність (густоту) розподілу цільового населення в досліджуваному регіоні та точних географічних координат фактичного розташування всіх діючих торговельних точок (магазинів і сервісних центрів) підприємства. Показники густоти та загальної кількості населення в розрізі адміністративних одиниць міста були зібрані та верифіковані нами за допомогою актуальних відкритих даних, що містяться в офіційних реєстрах та вільному доступі в мережі

Інтернет. Отриману демографічну статистику систематизовано у таблиці нижче.

Таблиця 3.2.

Кількість населення Києва за районами станом на 2025 рік (за даними доступних Інтернет-джерел)

	Наявне населення		Постійне населення	
	На перше червня 2025 року	Середня чисельність у січні-травні 2025 року	На перше червня 2025 року	Середня чисельність у січні-травні 2025 року
м.Київ	2928177	2926969	2886870	2885662
райони				
Голосіївський	251080	251003	248982	248905
Дарницький	337706	336884	331207	330385
Деснянський	368911	369067	366021	366177
Дніпровський	355419	355305	353690	353576
Оболонський	321229	321317	318355	318443
Печерський	156436	155764	151713	151041
Подільський	200664	200350	197254	196940
Святошинський	341305	341256	335385	335336
Солом'янський	370288	370197	368242	368151
Шевченківський	255139	225826	216021	216708

З іншого боку, збір точних географічних координат усіх існуючих місць продажу та обслуговування продукції фірми здійснювався за допомогою популярного картографічного сервісу Google Карти (Google Maps). Цей інструмент являє собою потужний комплекс спеціалізованих веб-програм, розроблених на базі безкоштовних світових сервісів картографії та передових ГІС-технологій корпорації Google. На сьогодні цей додаток масово використовується не лише для рутинного пошуку інформації на карті (з відмітками історичних пам'яток, організацій, вулиць чи станцій метрополітену), але й як професійний інструмент, що дозволяє оптимізувати маршрути кур'єрів, з високою точністю знаходити географічну відстань між об'єктами, а також швидко встановлювати точні GPS-координати (широту та довготу) розташування будь-якого об'єкта на поверхні Землі.

На першому етапі просторового аналізу, активно використовуючи

функціонал сервісу Google Карти, нами було візуально ідентифіковано та позначено на цифровій мапі всі діючі точки роздрібного продажу та фірмового сервісного обслуговування продукції компанії Sigma. Цей процес наочно проілюстровано на рисунку нижче (рис. 3.15).

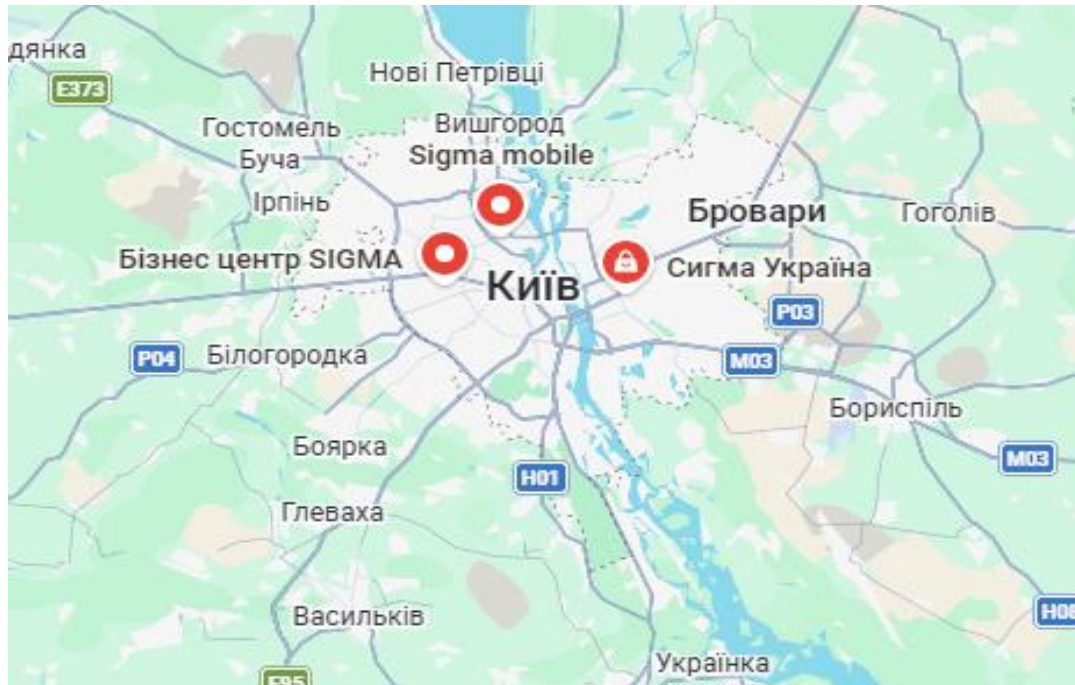


Рис. 3.15 – Визначення розташування торговельних точок фірми

Безпосереднє встановлення точних географічних координат розташування кожної окремої точки здійснювалося за чітким та простим алгоритмом. Спочатку оператор наводить курсор на потрібну локацію (будівлю магазину) на інтерактивній карті та натискає праву кнопку миші. Далі у контекстному меню, що з'явилося, необхідно вибрати спеціальний системний пункт «Що тут?». Відразу після цього у нижній інформаційній панелі екрана система автоматично генерує та відображає точні числові значення широти та довготи обраної точки (рис. 3.16).

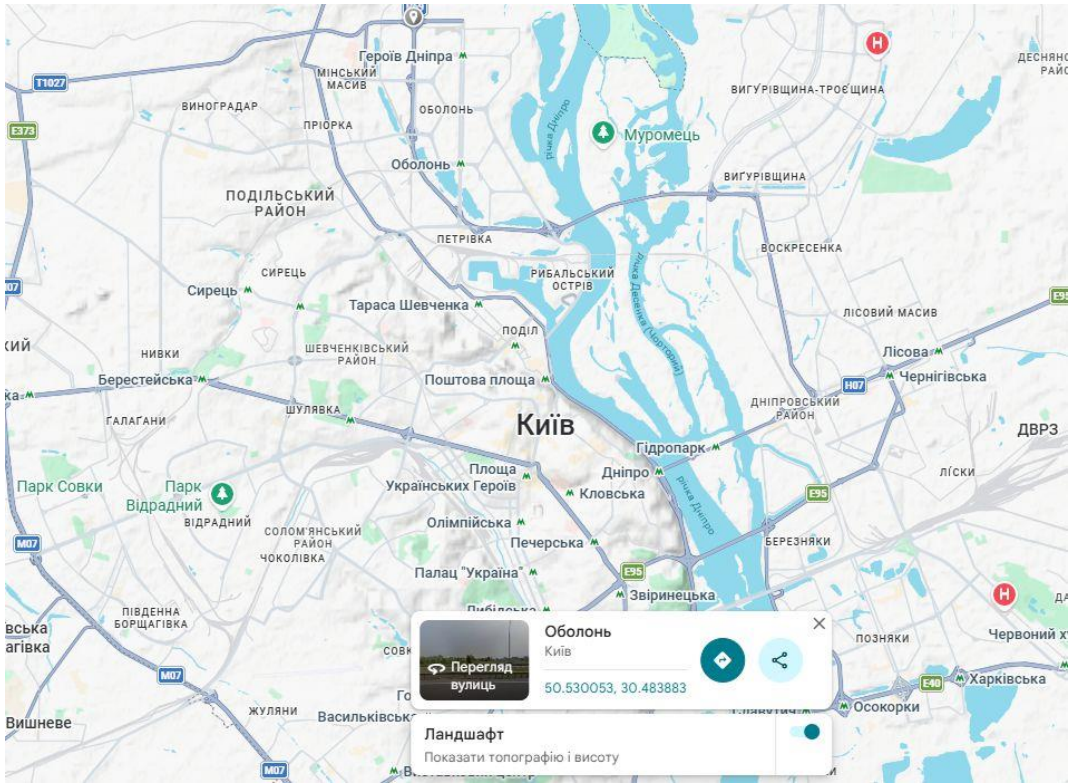


Рис. 3.16 – Визначення географічних координат торговельних точок

Усі вихідні геопросторові та демографічні дані, зібрані під час цього етапу дослідження, були ретельно перевірені, зведені та скомпоновані у єдину розрахункову матрицю (табл. 3.3). Ця таблиця стала фундаментальною базою для подальшого математичного моделювання оптимального логістичного вузла.

Таблиця 3.3.

Встановлення координат оптимального центру

Район	Вихідні дані		
	X	Y	H
Шевченківський	50,45	30,52	225139
Шевченківський	50,47	30,49	225139
Дніпровський	50,47	30,58	355419
Подільський	50,48	30,52	355419
Подільський	50,44	30,51	200664
Подільський	50,48	30,5	200664
Оболонський	50,49	30,53	321229
Оболонський	50,49	30,5	321229
Деснянський	50,45	30,63	368911
Печерський	50,43	30,55	200664
Центр	50,46729	30,53872	

Математичний розрахунок ідеальних координат для розміщення оптимального центру (головного складського приміщення) здійснювався за допомогою класичних формул визначення координат центру ваги (обслуговування). У цих формулах координати кожної торгової точки зважуються на показник чисельності населення відповідного району (що виступає в ролі «маси» або «ваги» вузла тяжіння):

$$x_u = \frac{\sum_{i=1}^{12} H_i x_i}{\sum_{i=1}^{12} H_i} \quad (3.3)$$

$$y_u = \frac{\sum_{i=1}^{12} H_i y_i}{\sum_{i=1}^{12} H_i} \quad (3.4)$$

Спираючись на результати проведених математичних обчислень, у середовищі табличного процесора MS Excel було побудовано точкову діаграму. Вона забезпечує наочне графічне представлення взаємного розташування всіх торгових точок та безпосередньо розрахованого оптимального місця для зведення чи оренди центрального складського приміщення (рис. 3.17).

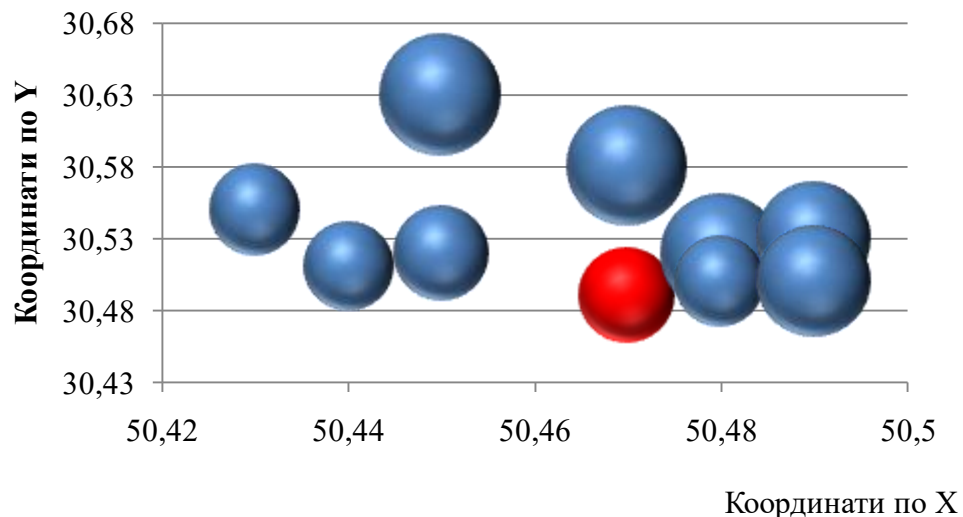


Рис. 3.17 – Графічне представлення оптимального місця розташування складського приміщення

Маючи на руках чіткі числові значення географічних координат, за

допомогою інструментарію пошуку в системі Google Карти було встановлено фізичну локацію розрахованої точки на реальній мапі міста. З'ясувалося, що математично ідеальна точка логістичного центру знаходиться у географічному центрі столиці — у Дніпровському районі, а саме на території Труханового острова. Хоча це місце є рекреаційною зоною, отриманий орієнтир дозволяє звузити радіус пошуку та почати аналіз реальних об'єктів комерційної нерухомості на прилеглих до цієї географічної точки промислових зонах (наприклад, на Подолі або лівому березі Дніпра) для подальшої довгострокової оренди складського приміщення (рис. 3.18).

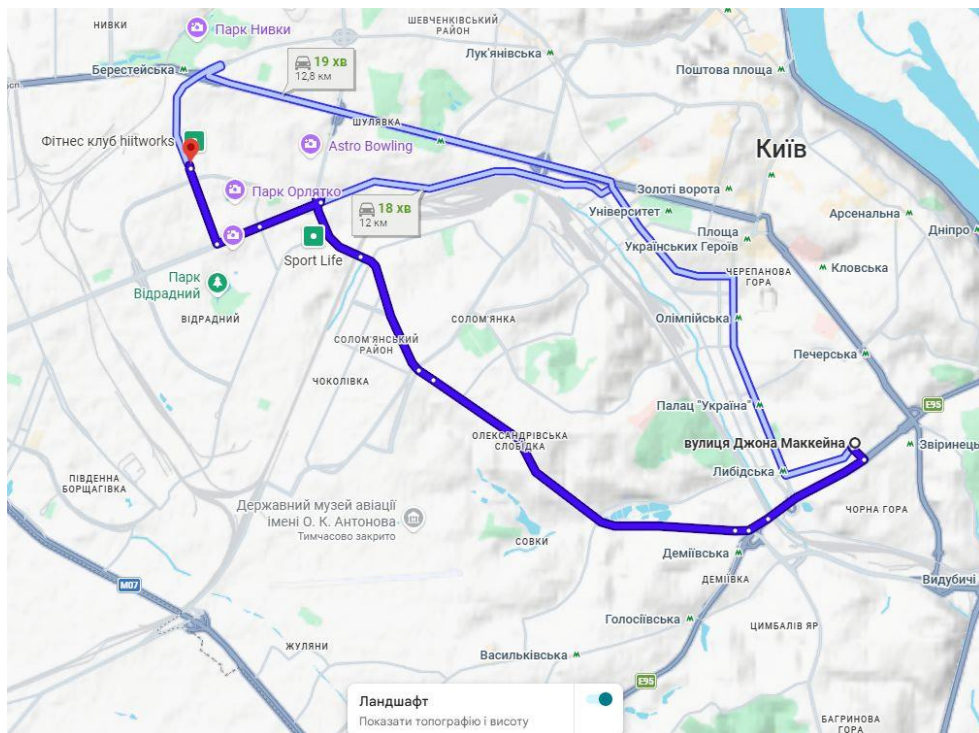


Рис. 3.18 – Пошук необхідного об'єкту за заданими координатами

Підсумовуючи результати цього етапу дослідження, можна констатувати надзвичайно важливий для сучасного менеджменту факт. Як наочно доводить наш практичний приклад, навіть базові та безкоштовні загальноновживані ІТ-засоби, такі як сервіс просторового аналізу Google Карти та стандартна офісна програма MS Excel, здатні забезпечити потужну аналітичну та ІТ-підтримку для прийняття складних стратегічних управлінських рішень. При цьому компанія абсолютно уникає додаткових грошових витрат на закупівлю спеціалізованого логістичного софту, не

потребує залучення дорогих сторонніх експертів у галузі ІТ або тривалого й вартісного навчання власних логістів роботі у складних корпоративних ПС-системах.

3.5. Технології баз даних у діяльності малого підприємства

Поряд із зазначеними вище можливостями оптимізації бізнес-процесів, фундаментальну роль у розбудові сучасних автоматичних та автоматизованих інформаційних систем відіграє впровадження передових технологій баз даних (БД). Практичне застосування таких технологій у корпоративному середовищі забезпечує виконання низки критично важливих функцій. Передусім, це гарантоване та надійне довгострокове зберігання колосальних обсягів структурованої інформації в пам'яті комп'ютерних систем. Крім того, технології БД дозволяють реалізовувати складні, вузькоспеціалізовані для конкретної галузі алгоритми обробки, фільтрації та швидкої передачі даних. Не менш важливою функцією є забезпечення співробітників максимально зручним, інтуїтивно зрозумілим та ергономічним графічним інтерфейсом для щоденної роботи з корпоративною інформацією [36].

З теоретичної точки зору, саме база даних виступає непорушним фундаментом для проектування будь-якої інформаційної системи. Під самим поняттям «інформаційна система» (ІС) у сучасній науці розуміють складний синергетичний комплекс, що складається безпосередньо з бази даних та сукупності апаратно-програмних засобів, які забезпечують безперебійне зберігання, гнучку модифікацію, миттєвий пошук потрібної інформації та комфортну взаємодію кінцевого користувача з машиною. Для ефективного керування цими масивами даних застосовується система управління базами даних (СУБД). Вона являє собою потужний комплекс спеціалізованих програмних і мовних засобів, розроблених виключно для створення, адміністрування, регулярного ведення та безпечного спільного використання бази даних усіма авторизованими співробітниками фірми. Головною та найбільш складною функцією СУБД є ефективне управління системною

пам'яттю. Зазвичай такі спеціалізовані програмні комплекси гарантують повну незалежність логічного рівня представлення даних від фізичного рівня їх зберігання на жорстких дисках [36].

Історичне прагнення розробників зробити архітектуру баз даних якомога гнучкішою та адаптивнішою призвело до появи та тотального домінування реляційної моделі даних. Вона запропонувала інженерам надзвичайно простий, але водночас високоефективний математичний механізм підтримки та контролю логічних зв'язків між різними масивами інформації. У межах цієї парадигми абсолютно всі дані концептуально представляються виключно у вигляді двовимірних таблиць. На сьогодні реляційна модель залишається єдиною з усіх існуючих, що здатна забезпечити строгу математичну однаковість та стандартизацію представлення будь-яких типів даних.

Спираючись на глибоко вивчений та проаналізований практичний досвід вітчизняних і закордонних компаній [51], з метою кардинального вдосконалення щоденної роботи менеджерів нашої фірми, нами було спроектовано та розроблено власну корпоративну базу даних під назвою «Менеджер». Цей інформаційний ресурс акумулює в собі вичерпну інформацію щодо поточних залишків товару на складах підприємства, актуальних цінових політик, детальних технічних характеристик кожної моделі пристрою, точних дат здійснення транзакцій, а також статистику частоти звернень до сервісного центру (регулярність поломок).

Фізичну реалізацію бази даних було здійснено на платформі популярної системи управління базами даних Microsoft Access. Вибір цього програмного продукту є цілком обґрунтованим: він є складовою частиною стандартного офісного пакету, а його вбудовані обчислювальні та архітектурні можливості здатні з величезним запасом задовольнити всі поточні інформаційні запити невеликої за чисельністю персоналу фірми.

Структурно база даних «Менеджер» складається з 6 взаємопов'язаних реляційних таблиць. У цих таблицях жорстко систематизовано всю первинну інформацію, яка безперервно надходить як від фахівців комерційного відділу (відділу продажів), так і від інженерів служби технічної діагностики. Для

забезпечення швидкої аналітики база даних містить 10 спеціально спроектованих запитів, використання яких радикально полегшує та пришвидшує цільовий пошук інформації. За допомогою низки цих запитів здійснюється автоматичне групування менеджерів за різними критеріями: загальним стажем роботи в компанії, кількістю успішно закритих угод, обсягом згенерованого прибутку тощо. Паралельно система здійснює складні фонові обчислення з подальшим виведенням на екран монітора відповідних КРІ-коефіцієнтів, що об'єктивно характеризують реальну ефективність діяльності кожного працівника. Згенерована аналітична інформація є надзвичайно затребуваною для HR-відділу та керівництва, оскільки слугує математичним підґрунтям для розробки прозорих мотиваційних заходів: нарахування індивідуальних надбавок, розрахунку преміальних фондів та підвищення лояльності персоналу.

З метою максимального полегшення процедури ручного введення нової інформації операторами, у системі передбачено 8 ергономічних електронних форм. Своєю чергою, швидке та зручне виведення підсумкової інформації на друк для керівництва забезпечують 8 автоматизованих звітів, створених за корпоративними стандартами компанії.

Процес взаємодії користувача з програмою починається з її запуску, під час якого на екрані з'являється інформативна стартова заставка. Вона містить стислу довідкову інформацію щодо основних структурних компонентів завантаженої бази даних, її головного призначення та вказівку на автора програмної розробки. Для економії робочого часу досвідчених користувачів розробники передбачили спеціальну опцію, яка дозволяє при подальших запусках програми не виводити цю заставку на екран (рис. 3.19).

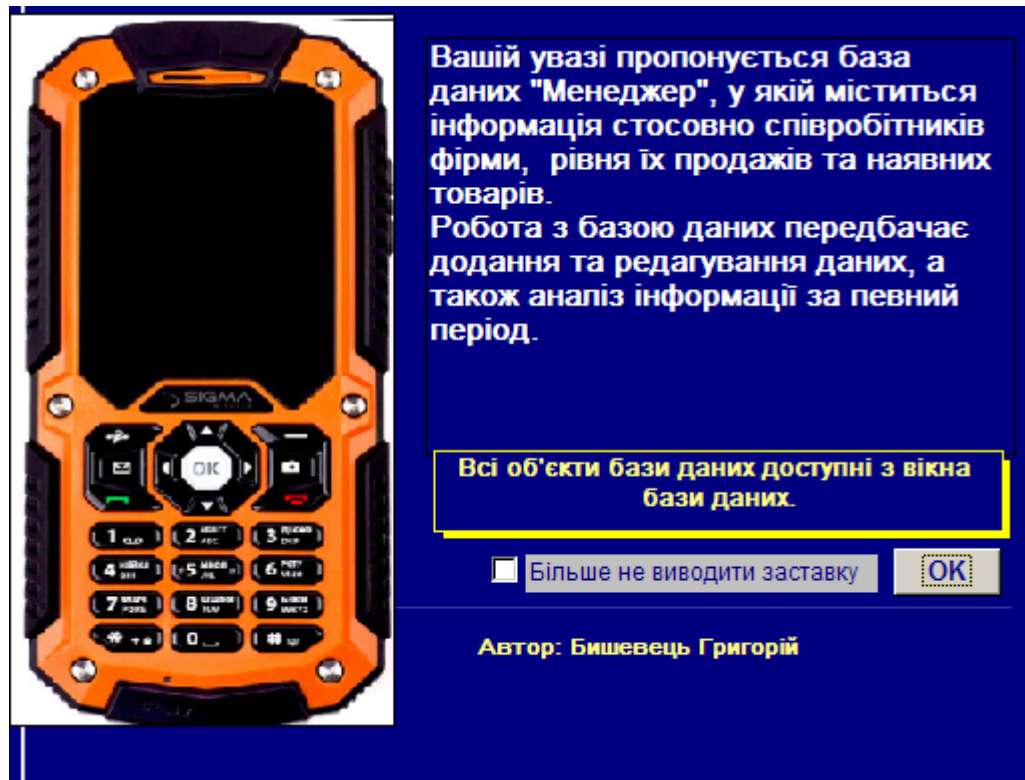


Рис. 3.19 – Заставка бази даних «Менеджер»

Відразу після ознайомлення із заставкою та натиснення на керуючу кнопку «OK», перед користувачем відкривається головна кнопкова форма (switchboard). Саме через цей центральний інтерфейс оператор може безперешкодно здійснювати всі необхідні робочі процедури: оперативне отримання довідкової інформації, додавання нових записів, редагування існуючих карток, а також складне багатокритеріальне сортування та пошук необхідних масивів даних. Слід особливо наголосити, що завдяки продуманій архітектурі головної форми, накопичену в базі даних інформацію надзвичайно зручно аналізувати. Керівник може в кілька кліків миші виявляти, які саме позиції товарів користуються ажіотажним попитом серед покупців, контролювати їх фактичну наявність на локальному складі, відстежувати моделі, що найчастіше потрапляють до сервісного центру, а також формувати рейтинг найбільш активних та результативних менеджерів із продажу техніки (рис. 3.20).



Рис. 3.20 – Головна кнопкова форма бази даних «Менеджер»

Упроваджена база даних надає потужний інструментарій для тотальної оптимізації процесу пошуку та обробки аналітичних даних. Для прикладу, просте натиснення на кнопку «Популярність товару» ініціює запуск складного SQL-запиту, в результаті якого всі наявні товари автоматично вишиковуються та відображаються у суворому порядку спадання частоти їх фактичних продажів. Аналогічно працює інструмент формування звітності: натиснення на кнопку «Продажі» запускає діалогову форму, в якій користувачеві достатньо лише внести початкову та кінцеву дати цікавого йому періоду. Система миттєво обробить запит і видасть деталізований звіт, що міститиме точні найменування та кількісні показники всіх товарів, реалізованих за вказаний проміжок часу.

Підсумовуючи, можна стверджувати, що розроблена реляційна база даних «Менеджер» характеризується такими критично важливими для бізнесу властивостями, як висока інформативність, цілодобова доступність та виняткова простота в повсякденному використанні. Вона надає користувачеві повний спектр можливостей для надійного зберігання, своєчасного

оновлення та гнучкого коригування початкових інформаційних масивів. Причому варто зробити особливий акцент на тому факті, що ефективна робота з цією базою даних абсолютно не вимагає від персоналу наявності глибоких спеціальних знань у галузі ІТ чи навичок програмування. Усі складні математичні та статистичні розрахунки проводяться «під капотом» безпосередньо логікою самої програми, що робить цей інструмент доступним для максимально широкого кола штатних фахівців різного профілю. Разом з тим, розробка та впровадження цієї системи також не вимагає від підприємства жодних додаткових капіталовкладень, оскільки СУБД MS Access традиційно входить до базового складу ліцензійного пакету офісних програм від корпорації Microsoft, яким уже користується компанія.

3.6. Шляхи удосконалення запропонованих засобів автоматизації роботи малого бізнесу

Розглядаючи подальші перспективи масштабування та вдосконалення вже запропонованих засобів автоматизації для малого підприємства, варто зосередитися на поглибленні їхнього аналітичного функціоналу. Зокрема, у комп'ютерній програмі «Стратегія» критично важливо програмно реалізувати повноцінний метод експертних оцінок, який науковці настійно рекомендують використовувати для точної кількісної оцінки якісних (суб'єктивних) даних. Відповідно до усталених рекомендацій фахівців у галузі статистики та кваліметрії, перевірка ступеня узгодженості думок членів експертної групи має здійснюватися традиційним математичним способом. Цей підхід передбачає вирішення класичної задачі аналізу даних експертизи за умови нестрогої послідовності ранжування експертних оцінок. Головним інструментом тут виступає коефіцієнт конкордації Кендалла (W), який в умовах наявності зв'язаних (однакових) рангів обчислюють за такою формулою:

$$W = \frac{12S}{m^2(n^3 - n) - m \sum_j^m \sum_{t_j} (t_j^3 - t_j)} \quad (3.5)$$

де базова змінна S відображає загальну суму квадратів відхилень усіх оцінок рангів кожного конкретного об'єкта експертизи від середнього арифметичного значення; параметр n позначає загальне число досліджуваних об'єктів експертизи; m — це фактична кількість залучених експертів; a розрахунковий показник T_j визначається через t_j — число однакових рангів, які присвоює різним альтернативам j -й експерт [5].

Паралельно з цим, для точного встановлення статистичної значущості отриманого коефіцієнта конкордації, алгоритм програми повинен автоматично розраховувати розрахункову величину критерію Пірсона χ^2 за наступною формулою:

$$\chi^2 = \frac{12S}{mn(n+1) - \frac{1}{n-1} \sum_{i=1}^m \sum_{t_j} (t_{ij}^3 - t_{ij})} \quad (3.6)$$

Після виконання цих обчислень отриманий результат програмно порівнюється з критичним (табличним) значенням для чітко заданого числа ступенів вільності $k = n - 1$ (що в нашому випадку дорівнює 11, оскільки оцінювалося 12 факторів) та на прийнятому в економічних дослідженнях рівні статистичної значущості $\alpha = 0,05$. Якщо розрахункове значення критерію виявляється меншим за табличне, то експертиза математично визнається такою, що не відбулася через критичну розбіжність у поглядах фахівців. Вочевидь, така заздалегідь встановлена та математично підтверджена узгодженість думок експертів допоможе керівництву надійно уникнути фатальних управлінських помилок безпосередньо в процесі стратегічного планування.

Окрім описаної раніше задачі пошуку оптимального географічного місця для оренди складського приміщення, існує величезний пласт інших економічних завдань. Значну частину щоденних логістичних викликів, що постають перед співробітниками фірми, надзвичайно доцільно розглядати як класичні оптимізаційні задачі потокового програмування та оптимального планування. Процес їхнього вирішення можна легко автоматизувати за допомогою вбудованої надбудови «Пошук рішень» у середовищі MS Excel. До таких комплексних завдань у першу чергу варто віднести оптимізацію

багатокритеріального управління товарними запасами на складах, мінімізацію витрат на транспортування (класична транспортна задача) та планування графіків постачання. Впровадження таких аналітичних інструментів сприятиме кардинальному вдосконаленню та здешевленню роботи служби логістики.

Розроблена нами корпоративна база даних «Менеджер» також містить величезний потенціал для подальшого розширення свого функціоналу. Спираючись на рекомендації фінансових фахівців, архітектуру цієї бази даних слід модернізувати таким чином, щоб її можна було застосовувати для автоматизованої підготовки платіжних відомостей, нарахування заробітної плати співробітникам та для оперативного вирішення інших суміжних питань базового бухгалтерського обліку й кадрового діловодства. Разом з тим, стратегічно необхідним кроком є розширення системи шляхом створення повноцінної клієнтської бази даних (початкового модуля CRM-системи). Це дозволить відділу маркетингу автоматизувати процес просування нових комерційних пропозицій вже існуючим клієнтам, організовувати цільові розсилки та ефективніше залучати нових зацікавлених осіб.

Що стосується авторської розробки «Діагностика гаджетів», то її життєвий цикл та подальше вдосконалення органічно пов'язані з глобальним розвитком екосистеми Google. Оскільки ресурс базується на хмарних технологіях, він буде автоматично удосконалюватися за рахунок постійного виходу оновлень, розширення функціоналу, посилення безпеки та додавання нових аналітичних інструментів безпосередньо розробниками сервісу Google Форми.

Підсумовуючи, варто зазначити, що всі запропоновані у роботі розробки жодним чином не вичерпують усе неосяжне розмаїття існуючих проблем та викликів у сфері автоматизації бізнес-процесів малих підприємств. Проте вони обґрунтовано розглядаються як потужні, дієві та необхідні перші кроки на шляху до глибокого технологічного вдосконалення діяльності фірми. Головна їхня цінність полягає у можливості здійснити цю цифрову трансформацію без залучення значних фінансових витрат та без стресової,

болісної реорганізації звичного ритму роботи співробітників компанії.

Висновки до розділу 3

У даному розділі було проведено глибоке теоретичне обґрунтування, а також здійснено безпосередню розробку та практичне впровадження комплексу доступних програмних продуктів, спрямованих на наскрізну автоматизацію роботи ключових підрозділів малого підприємства. На прикладі реальної компанії доведено ефективність застосування математичних моделей стратегічного менеджменту (програма «Стратегія»), хмарних технологій для збору технічної аналітики (веб-сервіс «Діагностика гаджетів»), оптимізаційних алгоритмів для логістики та реляційних систем управління базами даних (БД «Менеджер») для оптимізації комерційної діяльності. Запропоновані рішення довели свою економічну та операційну ефективність, а також було сформовано чіткий вектор їхнього подальшого масштабування та функціонального вдосконалення.

РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці при роботі за комп'ютером

Заходи щодо усунення небезпеки ураження електричним струмом зводяться до правильного розміщення устаткування та електричних кабелів. Інші заходи щодо забезпечення електробезпеки, збігаються з загальними заходами пожежо- та електробезпеки.

В якості профілактичних заходів для забезпечення пожежної безпеки слід використовувати скриту електромережу, надійні розетки з пожежобезпечних матеріалів, силові мережі живлення устаткування виконувати кабелями, розрахованими на підключення в 3-5 разів більшого навантаження, включати й виключати живлення обладнання за допомогою штатних вимикачів. Треба регулярно робити очистку внутрішніх частин комп'ютерів, іншого устаткування від пилу, розташовувати комп'ютери на окремих неспалюваних столах. Для запобігання іскріння необхідно рідше встромляти і виймати штепсельні вилки з розеток.

Екран дисплея повинен бути розташованим перпендикулярно до напрямку погляду. Якщо він розташований під кутом, то стає причиною сутулості. Відстань від дисплея до очей повинна трохи перевищувати звичну відстань між книгою та очима. Перед екраном монітора, особливо старих типів, повинен бути спеціальний захисний екран. При його відсутності треба сидіти на відстані витягнутої руки від монітора. Ще одним моментом, який стосується зору, є необхідність створення неоднорідного поля зору. Для цього можна розвісити на поверхнях (стінах) плакати та картини, виконані у спокійних тонах. Наприклад, пейзажі.

Важливою є форма спинки крісла, яка повинна повторювати форму спини. Висота крісла повинна бути такою, щоб користувач не почував тиску на куприк або стегна. Крісло бажано обладнати бильцями. Його потрібно встановити так, щоб не треба було тягтися до клавіатури. Періодично користувачу необхідно рухатися, вчасно змінювати положення тіла і робити

перерви у роботі.

При напруженій роботі за комп'ютером щогодини необхідно робити перерву на 15 хвилин через кожну годину і треба займатися іншою справою. Декілька разів на годину бажано виконувати серію легких вправ для розслаблення.

Наслідками регулярної роботи з комп'ютером без застосування захисних засобів можуть бути: захворювання органів зору (60% користувачів); хвороби серцево-судинної системи (20%); захворювання шлунково-кишкового тракту (10%); шкірні захворювання (5%); різноманітні пухлини.

Режим праці та відпочинку при роботі з персональною електронно-обчислювальною машиною (ПЕОМ) залежить від категорії трудової діяльності. Всі роботи з ПЕОМ ділять на три категорії. Перша - епізодичне зчитування і робота з інформацією не більше 2-х годин за 8-годинну робочу зміну. Друга - зчитування інформації або творча робота не більше 4-х годин за восьми годинну зміну. Третя - зчитування інформації або творча робота тривалістю більше 4-х годин за зміну.

Якщо у приміщенні експлуатується більше одного комп'ютера, то треба врахувати, що на користувача одного комп'ютера можуть впливати випромінювання від інших, в першу чергу бокових, а також і задньої стінки сусіднього дисплея. Тому необхідні спеціальними фільтрами і щоб користувач розміщався від бічних і задніх стінок інших дисплеїв на відстані не ближче одного метра.

Отже, щоб запобігти негативним впливам необхідно знати й небезпечні сторони самого комп'ютера і правила безпечної роботи, знати засоби запобігання небезпек. Вони пов'язані перед усім із загально відомими небезпечними факторами - поразками електричним струмом, пожежонебезпечністю.

Негативні фактори впливу на здоров'я. Всесвітня організація охорони здоров'я (ВООЗ) роботу з персональним комп'ютером віднесла до небезпечних, бо їй притаманний фактор постійно діючого стресу. Через це небезпеці піддаються всі життєво важливі органи людини, з'являється ризик

виникнення серйозних хвороб.

Електромагнітні поля біля комп'ютера (особливо низькочастотні) негативно впливають на людину і в першу чергу на її центральну нервову систему, викликаючи головний біль, запаморочення, нудоту, депресію, безсоння, відсутність апетиту, виникнення синдрому стресу. Причому нервова система реагує навіть на короткі за тривалістю впливи слабких полів: змінюється гормональний стан організму, порушуються біоструми мозку. Це призводить до погіршення зору, ускладненню серцево-судинних захворювань, зниженню імунітету, виникають негативні впливи на плин вагітності.

Характерною рисою професії оператора ПК є статичний режим роботи: великий обсяг праці треба виконувати в сидячому положенні. При цьому більшість груп м'язів постійно напружені, що призводить до швидкої стомлюваності, сприяє розвитку фахових патологічних вигинів хребта: грудному гіперкифозу, сплюсненню шийного лордозу і формуванню сколіозів. Неправильне розташування дисплеїв по висоті - занадто низьке або високе, під неправильним кутом - є головною причиною появи сутулості. Занадто високе розташування дисплея призводить до тривалої напруги шийного відділу хребта, що, зрештою, може призвести до розвитку остеохондрозу. Ненормальний стан хребта може стати причиною захворювання всього організму.

Основні "комп'ютерні" хвороби. Нерухома напружена поза оператора призводить до втоми і виникнення болю в хребті, шії, плечових суглобах. Інтенсивна робота з клавіатурою викликає болючі відчуття в ліктьових суглобах, передпліччях, зап'ястях і пальцях рук.

У деяких операторів персонального комп'ютера (ПК) розвивається м'язова слабкість, відбувається зміна форми хребта (синдром тривалого статичного навантаження - СТЧН), що може призвести до непрацездатності. Постійні користувачі ПК найчастіше піддаються психічним стресам, хворобам серцево-судинної системи і верхніх дихальних шляхів. Значному навантаженню піддається зоровий апарат.

При тривалій та інтенсивній роботі за комп'ютером з'являється синдром комп'ютерного стресу (СКС), який проявляється головною білью, запаленням очей, алергією, дратівливістю, млявістю і депресією, погіршенням зосередженості і працездатності.

Причинами різноманітних симптомів СКС є 5 основних чинників: неправильна робота очей і поза тіла, носіння невідповідних окулярів або контактних лінз, неправильна організація робочого місця, розподілення фізичних, розумових, візуальних навантажень, низький рівень візуальної підготовленості для роботи з комп'ютером. Особливо це характерно для дітей, молодших школярів.

Існує медико-соціальна проблема - комп'ютер і здоров'я дітей. Тепер в світі існує потужна індустрія виробництва комп'ютерних ігор. Діти з великим задоволенням віддають цим гарним і захоплюючим, а часто і агресивним іграшкам свій вільний час. Діти в значно меншому ступені, чим дорослі, спроможні контролювати себе. їхня психіка дуже нестійка, тому надмірне захоплення комп'ютерними іграми може стати причиною дуже важких наслідків - розвивається підвищена збудженість, у школярів знижується успішність, діти стають примхливими, некерованими, перестають будь-чим цікавитися крім комп'ютера.

За своїм впливом на дитячий організм комп'ютерна гра подібна наркотику, тому необхідно дуже строго дозувати час комп'ютерних занять. Для організму дитини важливим є правильний розвиток опірно-рухового апарату, відхилення від норм розвитку якого є хвороби.

Інтенсивна і тривала робота з клавіатурою комп'ютера може стати джерелом важких професійних захворювань рук. Робота з клавіатурою є причиною 12% профзахворювань, викликаних повторюваними рухами.

Захворювання, пов'язані з "травмами повторюваних навантажень", пов'язані з хворобами нервів, м'язів і сухожилів. Наприклад тендит - запалення і набрякання сухожилів. Захворювання поширюється на кисті рук, зап'ястя, плечі.

Хвороба Де-Кервена - різновид тендита, при якій страждають

сухожилля, пов'язані з великим пальцем руки.

Тендосиновіт - запалення синовіальної оболонки сухожильної частини рук.

Тунельний синдром зап'ясного каналу - запалення медіального нерва руки з-за набряку сухожилів, синовіальної оболонки.

Для зменшення шкідливого впливу клавіатури фірма Microsoft розробила ергономічну клавіатуру, що своєрідною формою, конструкцією знижує навантаження на руки. Творці клавіатури сподіваються, що вона застрахує оператора від тунельного синдрому зап'ястного каналу.

Інший пристрій, який привертає до себе увагу фахівців в області ергономіки - маніпулятор типу "миша". На жаль, навіть найерго-номічніша клавіатура не може цілком вирішити проблему, оскільки причини захворювання "травми повторюваних навантажень" дотепер цілком не виявлені.

Найбільш актуальним є захворювання, яке одержало назву "інтернетоманія", а психологи назвали її маніакальною залежністю від віртуального світу глобальних мереж. Збуджені, почервонілі очі, високий ступінь нервового і фізичного виснаження, сльозоточиве зівання - лише деякі симптоми цієї хвороби.

Сіткоголіки, або інтернетоголіки відчують жагуче бажання знову і знову занурюватися у світ віртуальної реальності і довго не виходити з нього. На думку психологів інтернетоманія також руйнівна, як і алкоголізм або наркоманія. Вона веде до глибоких змін особистості, самоізоляції, втраті внутрішніх орієнтирів, неурівноваженості психіки, що зовні часто проявляється розсіяністю і неохайністю, навіть байдужим відношенням до близьких.

Безпека в інтернеті. Сучасний інформаційний ринок є одним із найбільш яскравих досягнень сучасного світу. Одним з найширших інформаційних каналів, що дозволяє досягти цього, є світова комп'ютерна мережа Інтернет.

Початок створенню її поклав у 1972 р. Вінтон Серф - керівник лабораторії телекомунікацій в Стендфордді (США). Прототип такої системи

був створений, під назвою DARPA (по перших буквах одного з підрозділів міністерства оборони США). Головні конструктивні принципи, покладені в її основу, дозволили швидко охопити Сполучені Штати Америки (в першу чергу університети), а потім і весь світ. На початку 80-х років з'явилася нова назва системи -Internet (Всесвітня павутиння). В 1989 р. в Інтернеті налічувалося біля ста тисяч комп'ютерів, 1994 році їх було біля трьох мільйонів, в 2005 - біля 100 мільйонів.

Широкому поширенню Інтернету сприяло різноманіття засобів, які дозволяють людям з різних країн і континентів безперешкодно обмінюватися величезними обсягами інформації, що призвело до своєрідного інформаційного перевороту. Останнім часом одержали розвиток спеціалізовані сервери, присвячені питанням права і безпеки.

Таким чином, боротьба з економічною злочинністю переноситься у віртуальний (нереальний) світ, тобто світ створений комп'ютером, даючи можливість бізнесменам, приватним охоронцям, детективам і представникам правоохоронних органів обмінюватися інформацією і користуватися послугами інформаційних агентств, що працюють в цій сфері.

Комп'ютерні тенета охопили великі обсяги виробництва, банківську справу, наукові дослідження, освіти - від середньої до вищої. За допомогою глобального інтернету спілкується населення Землі. Виникає єдиний організм з спільною економікою, культурою, політикою, наукою, спільним інформаційним полем. Надалі ці інтегруючі процеси будуть наростати, але це може мати і негативні наслідки, зокрема з огляду загальної, індивідуальної безпеки.

Отже, щоб цьому запобігти потрібна висока комп'ютерно-інформаційна грамотність, і перед усім спеціалістів у будь-якій сфері діяльності - промисловій, науковій, навчальній.

4.2 Безпека в надзвичайних ситуаціях

Небезпека – це явища, процеси, об'єкти, інформація і самі люди, які можуть викликати небажані наслідки і призводити до погіршення стану

здоров'я чи смерті людини, завдавати шкоди навколишньому середовищу й об'єктам господарської діяльності. Життєвий досвід людини показує, що шкоду людині може нанести будь-яка діяльність: робота на виробництві (трудова діяльність), різні види відпочинку, розваги та навіть діяльність, пов'язана з навчанням.

Джерелами небезпек є природні процеси і явища, техногенне середовище і людські дії.

Небезпека об'єктивно існує в просторі і часі та реалізується у вигляді потоків енергії, речовини, інформації.

Здебільшого небезпека має прихований характер і може перетворюватися в реальну небезпеку за наявності таких умов:

- небезпека реально існує;
- людина перебуває в зоні дії небезпеки;
- людина не має ефективних засобів захисту, не використовує їх або ці засоби неефективні.

Усяка діяльність людини є потенційно небезпечною. Потенційна небезпека – це така небезпека, яка має прихований характер і може перетворитися в реальну небезпеку за наявності трьох умов, зазначених вище. Потенційна небезпека може реалізуватися у формі хвороб або травм. Але наявність потенційної небезпеки не завжди супроводжується її негативним впливом на людину. Для реалізації негативного впливу небезпеки необхідне виконання всіх трьох умов.

Умови, за якими небезпека може реалізуватися в подію, називаються небезпечною ситуацією.

Основними критеріями переходу небезпеки в надзвичайну ситуацію є:

- наявність або загроза загибелі людей чи значне погіршення умов їх життя.
- заподіяння економічних збитків.
- істотне погіршення стану довкілля.

Під час експлуатації ЕОМ, можуть мати місце наступні надзвичайні

ситуацій: пожежі, аварії теплопроводу, водопроводу, ураження електричним струмом.

Сам ЕОМ може стати причиною таких надзвичайних ситуацій як пожежа та ураження електричним струмом.

При виникненні надзвичайної ситуації користувачу необхідно негайно припинити роботу, вимкнути ЕОМ, інше електрообладнання і доповісти керівнику, якщо:

- помічено механічні пошкодження та інші дефекти електрообладнання і електропроводки;
- помітно підвищений рівень шумів при роботі на обладнанні;
- помітно підвищене тепловиділення від обладнання;
- постійно блимає екран монітора;
- стрибає текст на екрані монітора;
- відчувається запах горілого та диму;
- припинилася подача електроенергії.

Не приступати до роботи до повного усунення несправностей.

У випадку загорання або пожежі співробітники повинні негайно припинити роботу, відімкнути електроприлади, викликати пожежну команду, повідомити керівника робіт і приступити до ліквідації місця загорання, наявними засобами пожежогасіння.

При аварії водопроводу або тепломережі необхідно негайно визвати слюсаря-сантехніка, прибрати речовини та реактиви, які бурхливо реагують із водою, щоб на них не попала вода. Берегти від псування водою обладнання та прилади. Якщо усунути аварію неможливо, визвати міську аварійну службу.

При травмі у першу чергу звільнити постраждалого від травму чого фактора, повідомити керівника робіт, викликати медичну допомогу, надати першу долікарняну допомогу постраждалому і, по можливості, зберегти незмінною ситуацію до початку розслідування причин нещасного випадку.

4.3 Висновки до розділу

У цій частині магістерської роботи були викладені вимоги до робочого місця програміста. Створені умови повинні забезпечувати комфортну роботу. Дотримання умов, що визначають оптимальну організацію робочого місця програміста, дозволить зберегти гарну працездатність протягом усього робочого дня, підвищить, як у кількісному, так і в якісному відносінах продуктивність праці програміста.

ЗАГАЛЬНИЙ ВИСНОВОК

У межах проведеного дослідження було здійснено глибоке вивчення спеціалізованого аналітичного методу, спрямованого на суттєве підвищення точності ідентифікації та прогнозування загальної кількості прихованих помилок ще на ранніх стадіях життєвого циклу тестування. Застосування цієї методики наполегливо рекомендується для якісного вдосконалення та оптимізації процесу ухвалення управлінських рішень щодо того, чи досягла конкретна програмна система (ПС) заданого цільового рівня експлуатаційної надійності. Фундаментальною математичною основою цього методу виступає складний апарат нелінійного регресійного аналізу, який досліджує динаміку прогнозованої кількості дефектів у тісній прив'язці до розрахункового часу завершення процесу тестування.

Шляхом ретельного обчислення та інтерпретації параметрів отриманого рівняння регресії, інженери з якості отримують можливість сформувати абсолютно новий, максимально наближений до реальності та математично уточнений прогноз щодо залишкової кількості помилок у досліджуваному програмному продукті. Практичне впровадження та систематичне використання описаного аналітичного підходу дозволяє досягти відчутних результатів: або безпосередньо покращити точність кінцевого прогнозу на 3–5 %, або ж, за умови збереження базової точності, значно оптимізувати ресурси і скоротити загальну тривалість етапу тестування в середньому на 30–40 %.

Поряд із цим, варто наголосити, що одним із найбільш критичних та відповідальних завдань під час комплексного оцінювання показників надійності будь-якої програмної системи є не лише безпомилковий вибір найбільш адекватної математичної моделі надійності, але й процес збору репрезентативних вхідних статистичних даних, які формуються на основі реальних результатів тестування. Практика доводить, що цілеспрямоване підвищення якості таких первинних вхідних масивів інформації відіграє вирішальну роль. Йдеться, зокрема, про забезпечення максимальної

відповідності штучно створеного тестового профілю реальному операційному (експлуатаційному) профілю використання системи кінцевими користувачами. Крім того, надзвичайно важливим є досягнення максимально повного та глибокого покриття автоматизованими і ручними тестами всіх без винятку структурних модулів, логічних гілок алгоритмів, внутрішніх змінних програмної системи, а також визначених меж та класів їхньої еквівалентності. Виконання цих суворих умов дає потужну можливість багатократно підвищити загальну достовірність, об'єктивність та наукову обґрунтованість кінцевого оцінювання всіх метрик і показників надійності ПС.

У ході подальшого дослідження було детально вивчено та проаналізовано спеціалізовані інструментальні засоби, спеціально розроблені для безперервного оцінювання та динамічного прогнозування показників надійності програмних систем безпосередньо в процесі їхнього активного кодування та розроблення. Як вагомий практичний результат, було спроектовано, програмно реалізовано та успішно апробовано унікальний програмний засіб, головною функцією якого є глибоке попереднє опрацювання (парсинг та фільтрація) статистичних даних про зафіксовані відмови ПС. Цей інструмент органічно інтегрований із популярною корпоративною системою відстеження помилок Atlassian JIRA і фактично виконує роль потужного інтелектуального препроцесора для складної моделі надійності, що оперує специфічним показником архітектурної складності.

Крім того, у роботі детально описано ще один інноваційний програмний засіб, призначений для точного емпіричного оцінювання ймовірностей логічних переходів управління між різними модулями програмної системи. Цей модуль, написаний сучасною об'єктно-орієнтованою мовою програмування Java, функціонує на базі технології аспектно-орієнтованого реєстратора (логгера) поточного виконання ПС. У результаті роботи цього інструмента генеруються високоточні значення ймовірностей фактичного використання окремих модулів системи (що математично виражається у вигляді початкового вектора ймовірностей станів та квадратної матриці

ймовірностей переходів). Здобуті таким чином матричні та векторні дані в подальшому виступають надійним математичним фундаментом для проведення комплексного оцінювання загальної надійності програмного забезпечення з використанням передового компонентно-орієнтованого підходу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Барковський В. В. Застосування удосконалень метода найменших квадратів (МНК) до визначення закономірностей в сучасних телекомунікаційних мережах / В. В. Барковський, Т. В. Майсак // XII Міжнар. наук. конф. ім. академіка М. Кравчука : матеріали конф. (Київ, 15–17 трав. 2008 р.). – К. : Задруга, 2008. – С. 21.
2. Беркман Л. Н. Застосування вдосконалених модифікацій методу найменших квадратів до оптимізації телекомунікаційних мереж / Л. Н. Беркман, Т. В. Майсак // Зв'язок. – 2008. – № 7–8. – С. 46–52.
3. Беркман Л. Н. Рівні конвергенції і їх реалізація – ситуація у світі і в Україні / Л. Н. Беркман, Т. В. Майсак // Зв'язок. – 2009. – № 1 (85). – С. 19–22.
4. Брагина Т. И. Информационная технология рискориентированного оценивания функциональности web-ориентированных систем / Т. И. Брагина, Г. В. Табунщик // Системи обробки інформації. Вип.2 (118). – 2014. – С. 245–252.
5. Волочій Б.Ю. Надійнісна модель відмовостійкої багатопроцесорної систем з відновленням працездатності програмного забезпечення / Б.Ю. Волочій, Л.Д. Озірковський, О.В. Муляк, М.М. Змисний, Т. І. Панський // Вісник НТУУ «КПІ». Серія Радіотехніка. Радіоапаратобудування. – 2013. – № 54. – С. 33 – 43.
6. Вишнівський В.В. Основи надійності та діагностики телекомунікаційних і радіотехнічних систем. Конспект лекцій підготовлено для самостійної роботи студентів вищих навчальних закладів / В.В. Вишнівський. – Київ: ННІТІ ДУТ, 2015. – 142 с.
7. Майсак Т.В. Оптимізація параметрів телекомунікаційних систем та мереж на базі економетричних методів: автореф. дис. канд. тех. наук: спец. 05.12.02 / Т.В. Майсак; Державний університет інформаційно-комунікаційних технологій. – Київ, 2009. – 16 с.
8. Математичні моделі та методи аналізу надійності радіоелектронних, електротехнічних та програмних систем: монографія / Ю.Я. Бобало, Б.Ю.

Волочій, О.Ю. Лозинський, Б.А. Мандзій, Л.Д. Озірковський, Д. В. Федасюк, С.В. Щербовських, В.С. Яковина. – Львів: Видавництво Львівської політехніки, 2013. – 300 с.

9. Муляк О.В. Вплив моделей надійності програмних засобів на показники надійності програмно-апаратних систем / О.В. Муляк, В. С. Яковина, Б.Ю. Волочій // Східно-Європейський журнал передових технологій. – 2015. – Том 4, № 9(76). – С. 53–57.

10. Нитребич О.О. Моделі та методи оцінювання надійності програмного забезпечення з урахуванням його архітектури: дис. канд.техн. наук : 01.05.03 : захищена 19.03.15 : затв. 30.06.15 / О.О. Нитребич; Нац. ун-т "Львівська політехніка". – Львів, 2015. – 153 с.

11. Притула А. В. Практично-орієнтовані методи викладання в галузі вбудованих систем/ Притула А. В., Пархоменко А. В., Табунщик Г.В // Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: зб. тез доп. VII міжнар. наук. – практ. конф. (Запоріжжя, 17–19 вересня 2014 р.). – Запоріжжя: ЗНТУ, 2014. – С. 216–217.

12. Сенів М.М. Система оцінювання та прогнозування надійності програмного забезпечення / Сенів М.М., Федасюк Д.В., Парфенюк Ю.І., Яковина В.С., Чабанюк Я.М. // Відбір і обробка інформації. – 2010. – № 33 (109). – С. 123–129.

13. Табунщик Г.В. Віддалений та віртуальний інструментарій в інжинірингу: монографія / Г.В. Табунщик, Т.І. Каплієнко. – Запоріжжя: ЗНТУ, 2015. – С. 154-204.

14. Табунщик Г. В. Інженерія якості програмного забезпечення: навч. посіб. / Табунщик Г. В., Кудерметов Р. К., Брагіна Т. І.; Запоріз. нац. техн. ун-т. – Запоріжжя: Дике поле, 2013. – 173 с.

15. Тимошенко Ю.О. Узагальнена модель негомогенного пуассонівського процесу для оцінювання надійності програмного забезпечення / Ю. О. Тимошенко, М. В. Дідковська // Проблеми програмування. – 2004. – № 2–3. – С. 480–489.

16. Федасюк Д.В. Метод побудови сценаріїв тестування програмного

забезпечення на основі аналізу його змінних / Федасюк Д.В., Яковина В. С., Сердюк П.В., Нитребич О.О. // Інформаційні технології та комп'ютерна інженерія. – 2014. – № 2 (30). – С. 50–58.

17. Франчук В.М. Web-орієнтовані навчальні комп'ютерні системи / В. М. Франчук // Актуальні проблеми методології та методики навчання фізико-математичних дисциплін: матеріали Міжнародної наукової конференції, 18-19 січня 2013 року. – К.: Вид-во НПУ імені М. П. Драгоманова, 2013. – С. 124-125.

18. Франчук В.М. Вибір системи управління вмістом сайту / В. М. Франчук, О.В. Галицький // Науковий часопис НПУ імені М.П. Драгоманова. Серія № 2. Комп'ютерно-орієнтовані системи навчання / Редрада. – К.: НПУ імені М.П. Драгоманова, 2014. – № 14(21). – С. 19-28.

19. Франчук В.М. Управління інформаційними ресурсами засобами web-орієнтованих комп'ютерних систем в освітніх закладах / В. М. Франчук, О.В. Галицький // Тези доповідей II Міжнародної науково-практичної конференції «Інформаційні технології в освіті, науці і техніці» (ІТОНТ - 2014): Черкаси, 24-26 квітня 2014 р. – У 2-х томах. – Черкаси: ЧДТУ, 2014. – Т.2. – С. 28-29.

20. Чабанюк Я.М. Побудова і дослідження моделі надійності програмного забезпечення з індексом величини проекту / Я.М. Чабанюк, В. С. Яковина, Д.В. Федасюк, М.М. Сенів, У.Т. Хімка // Інженерія програмного забезпечення. – 2010. – № 1. – С. 24–29.

21. Яковина В.С. Методи та засоби аналізу надійності функціонування програмного забезпечення з урахуванням етапів життєвого циклу: автореф. дис. доктора тех. наук: спец. 01.05.03 / В.С. Яковина; Національний університет "Львівська політехніка". – Львів, 2015. – 325 с.

22. Яковина В.С. Огляд основних підходів до аналізу надійності програмного забезпечення / В. Яковина, В. Смірнов // Вісник Національного університету "Львівська політехніка". Комп'ютерні науки та інформаційні технології. – 2011. – № 719. – С. 278–282.

23. Arras, P. E-learning environment for the remote study in material properties courses/ Arras, P., Kolot, Y., Galyna, T., Kozik, T // International Journal of

Computing, 12 (3), 2014, 233–238.

24. Henke K. Using Interactive Hybrid Online Labs for Rapid Prototyping of Digital Systems / G. Tabushchyk, H-D Wuttke, St. Ostendorff, T. Vietzke // REV2014 Conference 26–28 February 2014, Porto, Portugal, PP. 48–59.

25. Heiko Koziolk. Operational Profiles for Software Reliability / Heiko Koziolk // Dependability Engineering, volume 2 of Trustworthy Software Systems. – 2006. – P. 119–142.

26. Fedasyuk D.V. Variable state-based software usage-model based on its variables / D. Fedasyuk, V. Yakovyna, P. Serdyuk, O. Nytrebych // Econtechmod. – 2014. – Vol. 3, № 2. – P. 15–20.

27. Mirmomeni M. Fuzzy descriptor systems and spectral analysis for chaotic time series prediction / M. Mirmomeni, C. Lucas, M. Shafiee, B.N. Araabi, E. Kamaliha // Neural Computing and Applications. – 2009. – Vol. 18. – P. 991–1004.

28. Salmeron M. Parallel Computation of an Adaptive Optimal RBF Network Predictor / M. Salmeron, J. Ortega, C.G. Puntonet, M. Damas // LNCS. – 2003. – Vol. 2687. – P. 425–432.

29. Yakovyna V.S. Influence of the activation function of RBF neural network on software reliability time series prediction / V. Yakovyna, V. Ketsman // VIIIth International Conference on Perspective Technologies and Methods in MEMS Design MEMSTECH'2012, Polyana, April 18–21, 2012: Proceedings. – P. 91–92.

30. JTAG Boundary – Тестування сканування в MAX V пристроїв. – Режим доступу: <http://www.altera.com/literature/hb/max-v/mv51008.pdf>

31. JTAG Boundary – Тестування сканування для Cyclone IV пристроїв: – Режим доступу: <http://www.altera.com/literature/hb/cyclone-iv/cyiv-51010.pdf>

32. Яковина В. С. Основи теорії надійності програмних систем / В. С.

Яковина, М. М. Сенів. – Львів : Видавництво Львівської політехніки, 2020. – 248 с.

33. Федасюк Д. В. Методи та засоби автоматизованого тестування програмного забезпечення / Д. В. Федасюк, В. С. Яковина // Вісник

Національного університету «Львівська політехніка». Серія : Комп'ютерні науки та інформаційні технології. – Львів : Вид-во Львів. політехніки, 2017. –

№ 864. – С. 112–119.

34. Харченко В. С. Верифікація та оцінювання безпеки інформаційно-керуючих систем / В. С. Харченко, О. А. Ілляшенко // Радіоелектронні і комп'ютерні системи. – Харків : НАКУ «ХАІ», 2018. – № 3(87). – С. 54–62.
35. Голего В. М. Основи технічної діагностики та надійності радіоелектронних об'єктів / В. М. Голего. – К. : НАУ, 2015. – 320 с.
36. Романюк О. Н. Інструментальні засоби опрацювання даних про відмови програмних систем на етапі тестування / О. Н. Романюк, Т. О. Савчук // Інформаційні технології та комп'ютерна інженерія. – Вінниця : ВНТУ, 2021. – № 2(51). – С. 76–84.
37. Волошин О. Ф. Оцінювання надійності програмного забезпечення на основі марковських моделей та компонентного підходу / О. Ф. Волошин, І. В. Козак // Проблеми програмування. – К. : Ін-т програмних систем НАН України, 2019. – № 1. – С. 22–31.
38. Стрілкова Т. В. Моделювання та оцінка надійності сучасних телекомунікаційних мереж / Т. В. Стрілкова, О. М. Ковальчук // Телекомунікаційні та інформаційні технології. – К. : ДУТ, 2018. – № 3. – С. 15–22.
39. Мельник А. О. Автоматизація промислових процесів на базі програмованих логічних контролерів / А. О. Мельник, О. І. Мороз // Автоматизація виробничих процесів у машинобудуванні та приладобудуванні – Львів: Вид-во Львів. політехніки, 2019.– Вип. 53.– С. 45–51.
40. Коваленко І. М. Застосування математичних пакетів для розрахунку показників надійності інфокомунікаційних систем / І. М. Коваленко, С. В. Сидоренко // Зв'язок. – К. : ДУТ, 2016. – № 4. – С. 38–43.
41. Петренко О. І. Метод нелінійного регресійного аналізу для прогнозування кількості помилок у програмних системах / О. І. Петренко, В. О. Мухін // Системні дослідження та інформаційні технології. – К. : ННК «ІПСА» НТУУ «КПІ», 2020. – № 4. – С. 89–98.