

**MODELING THE PROCESSES OF AN INFORMATION SYSTEM FOR ANALYZING AND CHECKING THE COMPOSITION OF PRODUCTS AND EVALUATING ITS QUALITY**

**МОДЕЛЮВАННЯ ПРОЦЕСІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ АНАЛІЗУ ТА ПЕРЕВІРКИ СКЛАДУ ПРОДУКТІВ ТА ОЦІНЮВАННЯ ЇЇ ЯКОСТІ**

**Anna Shilinh<sup>1</sup>**

DOI: <https://doi.org/10.30525/978-9934-26-355-2-12>

**Abstract.** *The purpose* of this paper is to model the processes of an information system for analyzing and checking the composition of products, as well as to evaluate the quality of this information system. A detailed description of the database structure of the information system for checking the composition of products, the requirements for its development and the relationship between its tables are also the tasks of this paper. The work includes modeling of the processes of the information system for analyzing and checking the composition of products. In particular, the interaction between the actors and entities of the information system is represented by a use case diagram. The main actors of the information system are identified as Anonymous User, Registered User and Administrator. The structure of interaction for the information system for analyzing and checking the composition of products is presented in the form of a relevant interaction diagram. The graphical representation of the set of actions of an information system is depicted using an activity diagram. Representation of information system's components, dependencies, and relationships between them is presented using a component diagram. The paper proposes to display computing nodes, components, and objects executed on these nodes using a deployment diagram. The paper also contains a selection of quality indicators for an information system for analyzing and checking the

---

<sup>1</sup> Candidate of Engineering Sciences, Associate Professor at the Department of Social Communications and Information Activity, Lviv Polytechnic National University, Ukraine

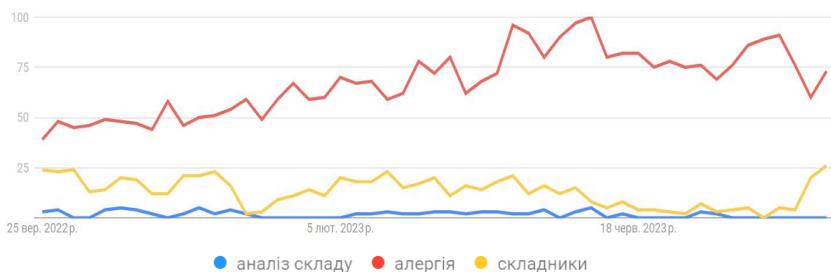
composition of products that satisfy the characteristics/sub-characteristics of software quality according to the ISO/IEC-25010 standard. The analysis of metrics that satisfy the characteristics/sub-characteristics of software quality according to the ISO/IEC-25010 standard shows that the following characteristics are suitable for evaluating the quality of an information system for analyzing and checking the composition of products: Functional suitability and its sub-characteristics Functional adequacy, Data exchangeability (Data format based), Data exchangeability, (User's success attempt based), as well as the characteristic Efficiency and its sub-characteristic Throughput. *Methodology of the study* is based on CASE-technologies for modeling the processes of an information system for analyzing and checking the composition of products, technologies and principles of database development. The study is also based on the selection of methods for evaluating the quality of an information system for analyzing and checking the product using ISO/IEC-25010 metrics that take into account the specifics of using this information system. *Results* of the quality indicators' monitoring of the information system for analyzing and checking the composition of products, which was carried out according to the defined indicators, showed that the information system is of appropriate quality according to the ISO/IEC-25010 standard, but identifies possible ways to improve it. *Practical implications*. The proposed model of an information system for analyzing and checking the composition of products and the structure of its database are the basis for developing the architecture of the relevant information system. The quality indicators defined in the work according to the ISO/IEC-25010 standard make it possible to establish appropriate quality criteria for similar information systems. *Value/originality* of an information system for analyzing and checking products is its versatility, in particular, its application not only in the food sector, but also in the cosmetic, medical and other industries.

## 1. Вступ

Сьогодні світ характеризується багатокomпонентністю у всіх сферах людського життя. Досягнення сучасної науки дають змогу поєднувати не лише компоненти у виготовленні різних продуктів, наприклад, харчових, косметичних, медичних, побутових засобів тощо, а також різні сфери наукової, соціальної та політичної діяльностей. Але люд-

ський організм володіє своєю індивідуальністю, а тому по-різному реагує на одні й ті самі продукти та засоби.

Аналіз рейтингів пошукових запитів користувачів віртуального простору за даними ресурсу <https://trends.google.com.ua/> за 2022–2023 рр. показує популярність інтересу з часом таких термінів як «аналіз складу», «алергія», «складники», що підтверджує актуальність дослідження. Показник інтересу з часом показує популярність пошукового терміну відносно найвищого значення для певного періоду часу. Причому, показник 100 – це пік популярності терміну.



**Рис. 1. Інтерес з часом пошукових термінів «аналіз складу», «алергія», «складники»**

Джерело: <https://trends.google.com.ua/> за 2022–2023 рр.

Саме тому виникає потреба аналізувати склад продуктів, які використовує людина у повсякденному житті. Отже, інформаційна система для аналізу та перевірки складу продуктів є ефективним та якісним засобом вирішення поставленого завдання.

Моделювання процесів інформаційної системи для аналізу та перевірки складу продукції та розробка відповідної структури бази даних є актуальним завданням на сьогодні. Зокрема, основні поняття технології проектування та адміністрування баз даних та сховищ даних є метою роботи [1]. У роботах [2; 3] розглядаються принципи проектування та моделювання програмного забезпечення сучасних інформаційних систем. Зокрема, у роботах [4; 5] визначено важливі принципами об'єктно-орієнтованого проектування інформаційних систем та їх багаторівнева архітектура. Застосування CASE-технологій як

основних засобів розробки програмних систем є предметом досліджень [6–8]. Дослідження [9] ґрунтується на аналізі методів та моделей проектування на основі сучасних CASE-засобів. Прикладне застосування CASE -технологій для розробки інформаційних систем є предметом досліджень [10–12].

Основні положення та метрики оцінювання програмного забезпечення за стандартом ISO/IEC-25010 викладені у документах [13–15].

Але жодне із досліджень не розглядає можливості моделювання процесів інформаційної системи аналізу та перевірки складу продуктів, що є основою розробки архітектури відповідної інформаційної системи, яка може бути адаптована до потреб будь-якої сфери діяльності людини, а також визначення показників для оцінювання цієї інформаційної системи. Саме тому, основним завдання цієї статті є розробка структури бази даних інформаційної системи для аналізу та перевірки складу продуктів, моделювання відповідних її процесів та оцінення якості цієї інформаційної системи на основі показників стандарту ISO/IEC-25010.

## **2. Специфікація вимог до інформаційної системи та структура бази даних**

Основним призначенням запропонованої інформаційної системи є отримання, обробка, зберігання та аналіз інформації щодо певного продукту. Ця інформаційна система дозволяє автоматизувати та пришвидшити процеси перевірки складу продуктів, збільшити точності та користь представленої користувачу інформації.

Саме тому, **основними завданнями інформаційної системи для аналізу та перевірки складу продукту є:**

- зручний та автоматизований процес аналізу складників вибраної продукції;
- якісна та швидка перевірка, що дозволить забезпечити економію часу та ресурсів користувача.

**Основними перевагами для користувачів цієї інформаційної системи є:**

- фіксування та огляд складу товарів з економією за допомогою сканування упаковки певного засобу;
- перевірка якості сканованого продукту;

– детальний опис складників, їх аналіз в комплексі та оцінку рівня безпеки товару;

– отримання інформації про продукти, які відповідають тим же вимогам користувача, але є кращими за якістю чи іншими показниками.

Структура бази даних інформаційної системи для аналізу та перевірки продуктів розроблена з дотриманням умов, які не допускають можливе накопичення непотрібних і недостовірних даних, збоїв в роботі, що може призвести до втрати інформації.

У кожній створеній таблиці бази даних є лише один первинний ключ, який слугує для ідентифікації рядків таблиці. Усі первинні ключі є цілими невід'ємними числами. Таблиці, які мають зовнішній ключ мають певне відношення із таблицею, де знаходиться цей зовнішній ключ.

Таблиця для зберігання даних авторизації містить унікальний атрибут, який відповідає за ідентифікацію користувача (Login), та зашифровує його пароль (Password). В цій таблиці також є зовнішній ключ, який буде набувати не нульове значення, якщо цей користувач є клієнтом сайту, а не його адміністратором.

Структура бази даних інформаційної системи для перевірки складу продукції представлена на рисунку 2.

Таблиця Users призначена для збереження даних авторизації усіх користувачів веб-сайту, типу користувача (клієнт чи адміністратор) та ідентифікатор клієнта, який є зовнішнім ключем для користувачів, що є клієнтами сайту. У таблиці Clients зберігається інформація про користувачів, а саме їх даних авторизації, ініціалів, дати народження та електронної пошти. Таблиця Allergies призначена для зберігання інформації про різні типи алергій (їх назви та опис).

У таблиці Products зберігається інформація про різні продукти (їх назви та опис). Таблиця Components містить відомості всіх існуючих компонентів, які додають до продуктів, а саме їх назви, опис та оцінка. Таблиця Search призначена для зберігання історії пошуку зареєстрованого користувача. Таблиця має містити такі атрибути як ідентифікатор користувача (зовнішній ключ), дата пошуку та змінну типу Boolean, яка відповідає за бажання користувача зберегти результат даного пошуку.

Між таблицями є відношення, які відповідають за залежності між ними. Побудовані такі відношення між таблицями:

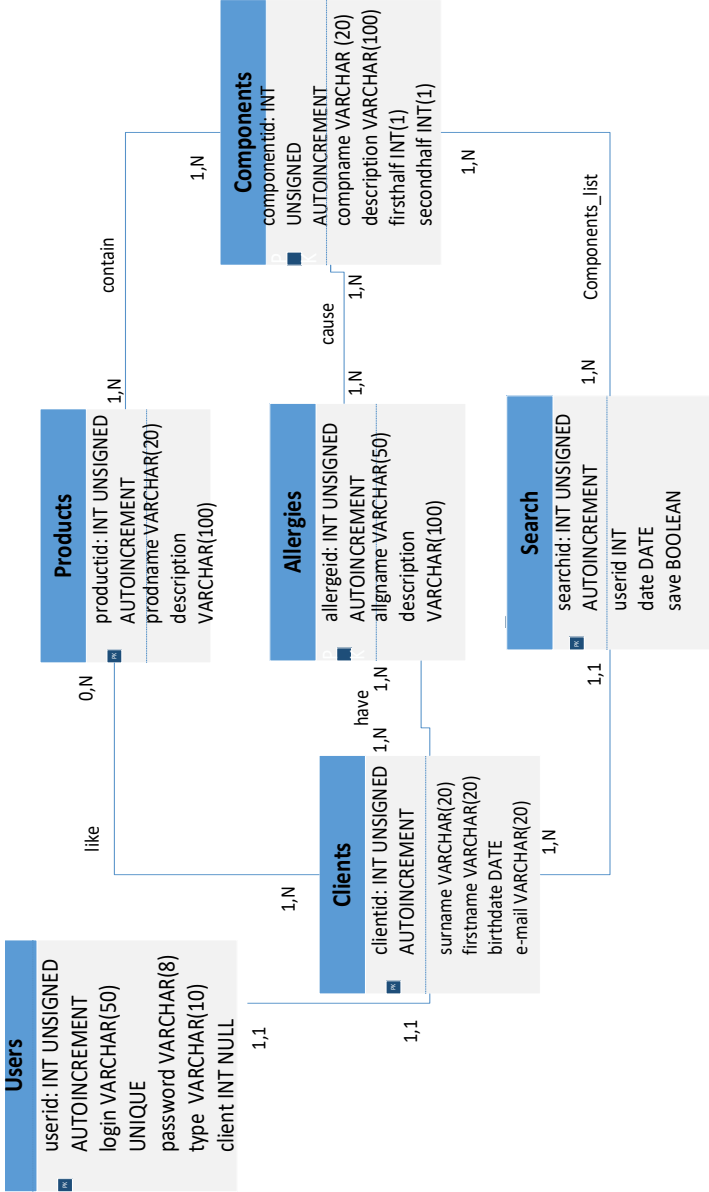


Рис. 2. Структура бази даних інформаційної системи для аналізу та перевірки складу продуктів

Джерело: власна розробка

– Users – Clients – відповідає за те, що ідентифікатор клієнта буде зовнішнім ключем у таблиці з користувачами. Відповідно зв'язок 1 до 1 означає, що лише один ідентифікатор може відповідати одному зовнішньому ключу.

– Clients – Products – відповідає за те, що користувач вподобав даний продукт і бажає додати його до списку вподобаних. Відповідно зв'язок 1 до N та 0 до N означає, що від 1 до багатьох користувачів можуть вподобати або багато різних продуктів або жодного. При цьому зв'язку створюється нова таблиця, яка буде містити відповідно ідентифікатори користувачів та продуктів.

– Clients – Allergies – відповідає за те, що користувач має алергію або їх декілька. Відповідно зв'язок 1 до N означає, що від 1 до багатьох користувачів мають від 1 до багатьох різних алергій. При цьому зв'язку створюється нова таблиця, яка буде містити відповідно ідентифікатори користувачів та алергій.

– Products – Components – відповідає за те, що продукт містить відповідну кількість компонентів. Відповідно зв'язок 1 до N та 1 до N означає, що від 1 до багатьох продуктів містять від 1 до багатьох компонентів. При цьому зв'язку створюється нова таблиця, яка буде містити відповідно ідентифікатори продуктів та компонентів.

– Components – Allergies – відповідає за те, що різні компоненти спричиняють різні алергії. Відповідно зв'язок 1 до N та 1 до N означає, що від 1 до багатьох компонентів спричиняють від 1 до багатьох алергій. При цьому зв'язку створюється нова таблиця, яка буде містити відповідно ідентифікатори компонентів та алергій.

– Search – Components – відповідає за те, що при конкретному пошуку здійснюється введення списку із компонентами продукту. Відповідно зв'язок 1 до N та 1 до N означає, що від 1 до багатьох пошуків можуть мати від 1 до багатьох компонентів. При цьому зв'язку створюється нова таблиця, яка буде містити відповідно ідентифікатори пошуку та компонентів.

– Clients – Search – відповідає за те, що ідентифікатор користувача є зовнішнім ключем в таблиці пошуків. При цьому, і користувач може мати багато різних пошуків, але 1 пошук може бути здійснений лише 1 відповідним користувачем.

### 3. Моделювання процесів інформаційної системи

Основою для проектування інформаційних систем на сьогодні є технології на основі CASE-засобів [2; 7]. Саме тому модель функціонування інформаційної системи для аналізу та перевірки складу продуктів представлено за допомогою UML-діаграм варіантів використання, кооперації, діяльності, компонентів та розгортання.

Діаграма варіантів використання відображає взаємодії між акторами та сутностями інформаційної системи для аналізу та перевірки складу продукту. Ця взаємодія є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання (рис. 3).

Для даної інформаційної системи визначено три актори: Анонімний користувач, Зареєстрований користувач та Адміністратор.

Перелік варіантів використання для:

– **Анонімного користувача:**

- зареєструватися;
- вписати/обрати компоненти;
- здійснювати аналіз;
- отримати рекомендації;
- переглянути додаткову інформацію.

– **Зареєстрованого користувача** (володіє всіма варіантами використання Анонімного користувача і додатково):

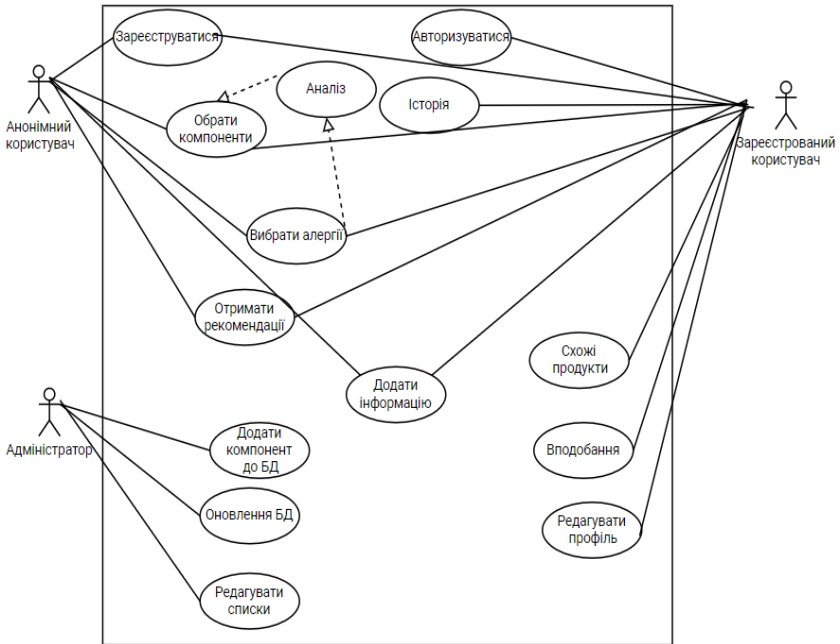
- авторизуватися;
- переглянути історію;
- переглянути схожі продукти;
- додавати продукт до вподобаних;
- редагувати профіль.

– **Адміністратора:**

- додавати компоненти/продукти до бази даних;
- оновлювати дані в базі даних;
- редагувати списки складників, алергій та відповідних рекомендацій.

Структура взаємодії для інформаційної системи для аналізу та перевірки складу продукту відображена за допомогою діаграми кооперації. Це дозволяє специфікувати особливості реалізації окремих найбільш





**Рис. 3. Діаграма варіантів використання інформаційної системи аналізу та перевірки складу продуктів**

*Джерело: власна розробка*

значущих операцій в інформаційній системі. На рисунку 4 подано діаграму кооперацій для Анонімного користувача.

Отже, діаграма кооперації для Анонімного користувача, містить такі об'єкти:

- Анонімний користувач – об'єкт класу Користувачів;
- Аналіз продукту – об'єкт класу, який відповідає за аналіз продукту на основі обраних користувачем компонентів;
- Загальні списки – об'єкт, який відповідає за створення списків із обраних компонентів користувачем;
- Списки алергій – об'єкт, який відповідає за створення списків із доданих користувачем алергій;
- База даних – об'єкт, який відповідає за зв'язок із базою даних.



**Рис. 4. Діаграма кооперації для Анонімного користувача**

*Джерело: власна розробка*

Діаграма кооперації для Зареєстрованого користувача містить чотири об'єкти класів із діаграми для анонімного користувача та додаткові об'єкти, а саме:

- Профіль – об'єкт, який відповідає за операції, які здійснюються в профілі користувача;
- Зареєстрований користувач – об'єкт класу Користувачів;
- Вхід/Вихід – об'єкт, який відповідає за авторизацію користувача та перевірку авторизації та вихід із системи.

Діаграма кооперації для Адміністратора містить два об'єкти із діаграми для зареєстрованого користувача – Вхід\Вихід та База даних – та чотири інших об'єкти, а саме:

- Адміністратор – об'єкт класу Користувачів;
- Додати елемент – об'єкт, який відповідає за додавання ного елементу в базу даних;
- Редагувати елемент – об'єкт, який відповідає за модифікацію елементів в базі даних;

– Видалити елемент – об’єкт, який відповідає за видалення елемента в базі даних.

На початку роботи система після початкового стану переходить у стан очікування дії користувача.

Якщо користувач обирає зареєструватися або здійснити авторизацію, система переходить у стан розширеного функціонування. Стан розширеного функціонування має вкладені стани. Здійснюється перехід у стан відображення сторінки профілю. Звідси можливий перехід у стан додавання алергій, а також у стан створення спеціального списку. Зі стану створення спеціального списку можливий перехід у стан додавання або створення нового продукту у межах створеного спеціального списку.

Зі стану очікування дій користувача також можливо потрапити у стан відображення сторінки аналізу, після чого система повертається до стану очікування дій користувача. Якщо компоненти вводяться вручну, система переходить до стану вводу даних. Якщо компоненти або продукт обираються із загального списку, система переходить до стану виконання запиту до бази даних, а із нього – у стан виконання аналізу продукту. Якщо користувач є зареєстрованим та має доступ до розширеного функціоналу, у нього є можливість додати продукт зі спеціального списку. В такому випадку переходимо у стан додавання або створення нового продукту у межах спеціального списку, а далі – у стан виконання аналізу продукту.

Після переходу до стану виконання аналізу складу продукту, відбувається перехід до стану відображення результатів. Якщо користувач зареєстрований і вирішує додати продукт до улюблених, система переходить до відповідного стану, а згодом – до стану відображення схожих продуктів. Також з стану відображення результатів можливий перехід у відображення рекомендацій.

Також на початку користування системою зі стану очікування дії користувача можливий перехід у стан адміністрування за умови, що користувач входить у систему в якості адміністратора. Стан адміністрування містить вкладені стани. Перехід відбувається у стан додавання продукту або списку до бази даних, після цього відбувається перехід до стану оновлення бази даних.

Перехід до стану «Вихід» відбувається зі стану оновлення бази даних, або зі стану перегляду схожих продуктів, або із стану відображення рекомендацій. Стан «Вихід» є останнім перед фінальним станом.

Графічне зображення набору дій інформаційної системи представлено за допомогою діаграми діяльності. Це дозволяє описати паралельні та умовні дії, а також використати випадки та системні функції на детальному рівні. На рисунку 5 діаграма діяльності представлена у варіанті для Анонімного користувача.

Для Анонімного користувача система перевіряє наявність облікового запису та надає можливість зареєструватися. Якщо користувач обирає роль Анонімного користувача, то інформаційна система запропонує ввести, відомі для користувача алергії, або одразу аналізувати продукт. Аналіз продуктів можна аналізувати на основі його компонент або наявних продуктів у базі даних. Якщо необхідний компонент є відсутній у базі даних, то у користувача є можливість ввести його вручну. Після цього користувача отримує аналіз обраного продукту.

Зареєстрований користувач одразу вводить свої дані для авторизації у системі. Він також має можливість додати алергії до свого особистого списку, який зберігатиметься у базі даних інформаційної системи.

Зареєстрований користувач, так як і Анонімний користувач, може аналізувати та перевіряти продукт за допомогою його компонент, або наявної інформації про продукт у базі даних інформаційної системи.

Але, на відміну від Анонімного користувача, у нього є можливість додавати обрані продукти до свого особистого списку, який зберігається у базі даних інформаційної системи. Це значно пришвидшить аналіз та перевірку, якщо користувач захоче провести їх знову. Також Зареєстрований користувач може додати перевірений продукт до улюблених, отримати рекомендації та переглянути схожі за компонентами продукти.

Основними можливостями для Адміністратора інформаційної системи аналізу та перевірки продуктів є додавання нових продуктів та компонентів до бази даних інформаційної системи, редагувати уже існуючі списки продуктів та компонент, а також оновлення бази даних.

Відображення компонент інформаційної системи, залежності та зв'язки між ними представлено за допомогою UML-діаграми компонентів (рис. 6).

Вона включає перелік таких компонентів як:

– DataBase – база даних системи, яка містить інформацію про компоненти, продукти та дані користувачів;

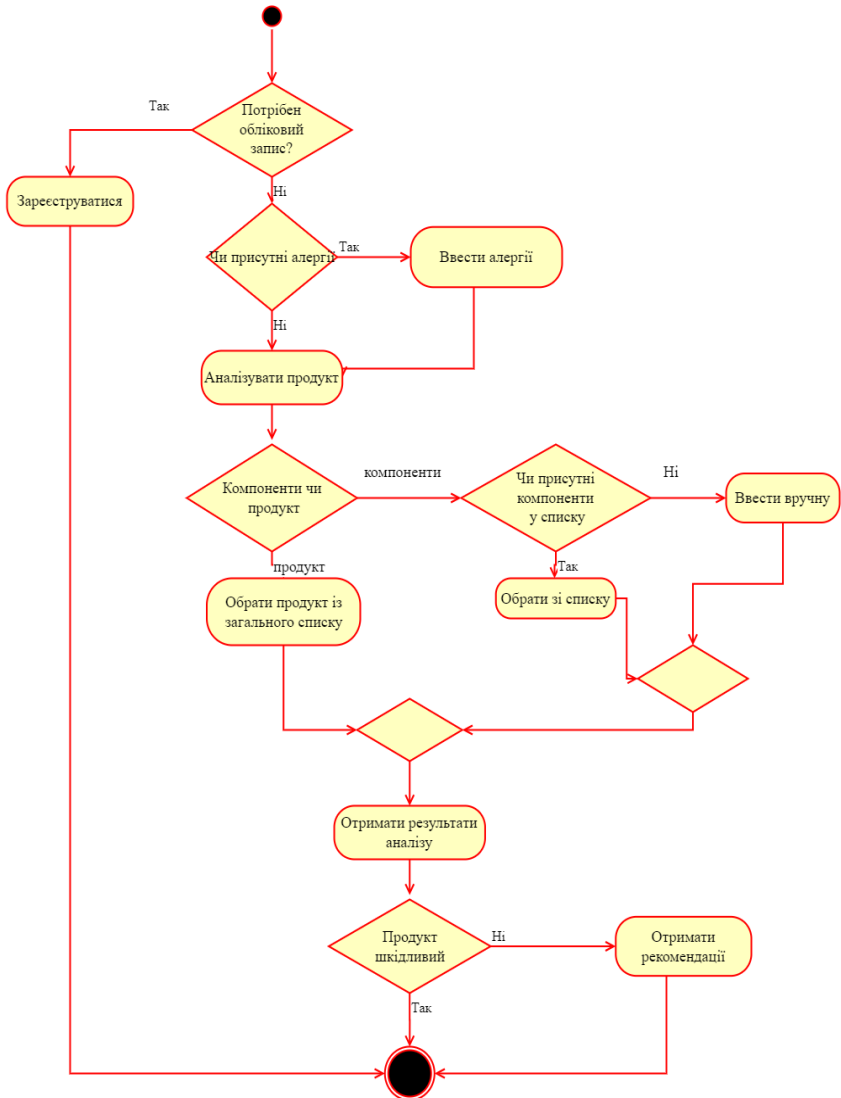


Рис. 5. Діаграма діяльності для Анонімного користувача

Джерело: власна розробка

- Main.exe – програма аналізу продуктів;
- UI;
- Спеціальний реєстратор винятків;
- Аналіз компонентів;
- Рекомендація схожих товарів.

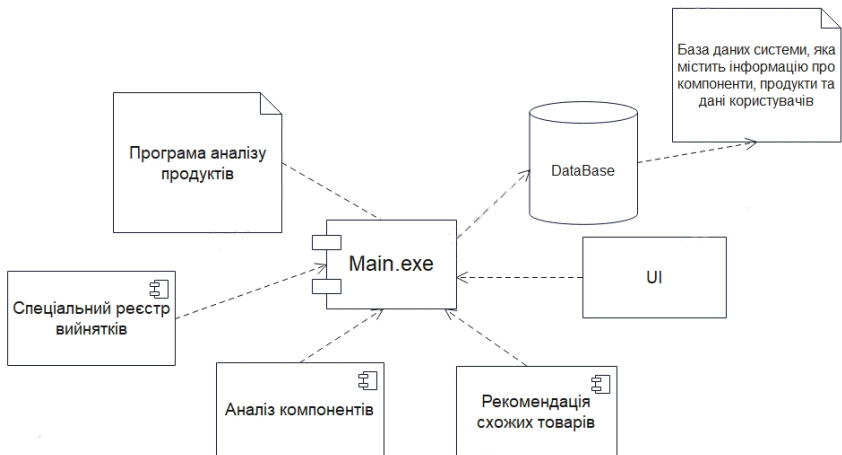
Відображення обчислювальних вузлів під час роботи інформаційної системи, компоненти, та об'єкти, що виконуються на цих вузлах представлено за допомогою UML-діаграма розгортання (рис. 7).

Діаграма розгортання містить два процеси та один пристрій.

До процесів відносимо:

- Пристрій – мобільний телефон або планшет при використанні камери, персональний комп'ютер або ноутбук.
- Сервер – процесор, який містить зв'язок із базою даних сайту і надсилає запити до неї, а також виконує всі запити користувача.

Пристрій – Мережа – закрита мережа, через яку здійснюється надсилання даних між пристроєм користувача та сервером, відповідно має з'єднання із вузлами Пристрій та Сервер.



**Рис. 6. Діаграма компонентів інформаційної системи для аналізу та перевірки складу продукту**

*Джерело: власна розробка*



**Рис. 7. Діаграма розгортання інформаційної системи для аналізу та перевірки складу продукту**

*Джерело: власна розробка*

#### **4. Вибір показників для оцінки якості інформаційної системи**

Для оцінювання якості інформаційної системи для аналізу та перевірки складу продуктів застосовано характеристики/під-характеристики якості програмного забезпечення за стандартом ISO/IEC-25010.

Аналіз метрик, що відповідають характеристикам/під-характеристикам якості програмного забезпечення за стандартом ISO/IEC-25010, показує, що для оцінювання якості інформаційної системи для аналізу та перевірки продуктів придатними є такі характеристики/під-характеристики:

##### **1) Функціональність (Functional suitability):**

- Функціональна адекватність (Functional adequacy);
- Здатність до взаємодії на основі форматів даних (Data exchangeability (Data format based));
- Функціональна перевірка доступу (Data exchangeability (User's success attempt based))

##### **2) Ефективність (Efficiency):**

- Ефективність за пропускнуою здатністю (Throughput).

Вимоги щодо оцінювання якості інформаційної системи для аналізу та перевірки продуктів задовольняють таку під-характеристику Функціональності як Функціональна адекватність (Functional adequacy), Здатність до взаємодії на основі форматів даних (Data

exchangeability (Data format based)), Функціональна перевірка доступу (Data exchangeability (User's success attempt based)).

Під-характеристика Функціональна адекватність (Functional adequacy) – це міра, за якої програмний продукт містить певну кількість функцій для виконання поставлених завдань, порівняно з кількістю необхідних функцій, які визначені технічним завданням.

Цю відповідність визначається наступним чином:

$$Q^{(1.1)} = 1 - \frac{A^{(1.1)}}{B^{(1.1)}}, \quad (1)$$

де  $A^{(1.1)}$  – кількість функцій інформаційної системи з виявленими проблемами,  $B^{(1.1)}$  – загальна кількість функцій інформаційної системи визначена технічним завданням.

Таким чином Функціональна адекватність показує функціональну відповідність розробленої інформаційної системи до технічного завдання.

Під-характеристика Здатність до взаємодії на основі форматів даних (Data exchangeability (Data format based)) – це міра, за якої програмний продукт містить кількість форматів даних, дозволених для обміну з іншим програмним забезпеченням або системою, порівняно із загальною їх кількістю.

Цю відповідність визначаємо так:

$$Q^{(1.3)} = \frac{A^{(1.3)}}{B^{(1.3)}}, \quad (2)$$

де  $A^{(1.3)}$  – кількість форматів даних, успішно використаних для обміну даними з іншим програмним забезпеченням чи системою,  $B^{(1.3)}$  – загальна кількість форматів даних, що розглядаються у технічному завданні для інформаційної системи.

Таким чином Здатність до взаємодії на основі форматів даних (Data exchangeability (Data format based)) показує здатність інтеграції розробленої системи у процесі взаємодії з іншим програмним забезпеченням або системою відповідно до технічного завдання.

Під-характеристика Функціональна перевірка доступу (Data exchangeability (User's success attempt based)) – це міра, за якої програмний продукт містить кількість успішних спроб доступу до інформаційної системи.



Ця відповідність визначається наступним чином:

$$Q^{(1.4)} = \frac{A^{(1.4)}}{B^{(1.4)}}, \quad (3)$$

де  $A^{(1.4)}$  – кількість спроб доступу користувачів до системи та даних, які записані у базі даних історії доступу,  $B^{(1.4)}$  – кількість спроб доступу користувачів до системи та даних, здійснених під час оцінки.

Таким чином Функціональна перевірка доступу Data exchangeability (User's success attempt based) показує рівень доступу користувачів до розробленої системи відповідно до технічного завдання.

Вимоги щодо оцінювання якості інформаційної системи для аналізу та перевірки продуктів задовольняють таку під-характеристику ефективності (Efficiency) як Ефективність за пропускну здатністю (Throughput).

Під-характеристика Ефективність за пропускну здатністю (Throughput) – це міра, за якої програмний продукт містить кількість успішно виконаних завдань інформаційної системи за певний проміжок часу.

Цю відповідність визначаємо так:

$$Q^{(4.1)} = \frac{A^{(4.1)}}{T}, \quad (4)$$

де  $A^{(4.1)}$  – кількість виконаних завдань,  $T$  – період часу спостереження.

Таким чином Ефективність за пропускну здатністю (Throughput) показує рівень ефективності інформаційної системи у визначений проміжок часу відповідно до технічного завдання.

## 5. Аналіз результатів

Моніторинг показників якості інформаційної системи для аналізу та перевірки складу продуктів здійснювався за такими показниками:

- Функціональна адекватність (Functional adequacy);
- Здатність до взаємодії на основі форматів даних (Data exchangeability (Data format based));
- Функціональна перевірка доступу (Data exchangeability (User's success attempt based))
- Ефективність за пропускну здатністю (Throughput).

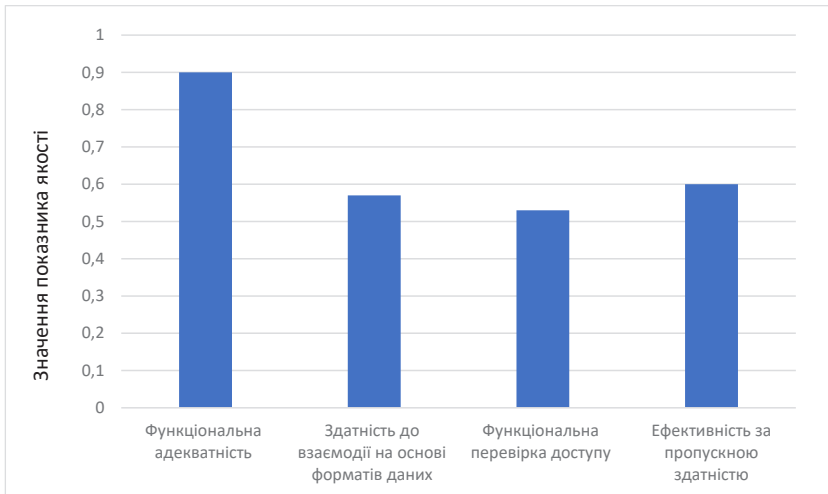
Для аналізу динаміки показників якості інформаційної системи аналізу та перевірки складу продуктів вважатимемо значення показ-

ника оптимальним, якщо воно належить проміжку  $[0,7; 1)$ , і прийнятним, якщо воно належить проміжку  $[0,4; 0,7)$ .

Значення показників якості інформаційної системи для аналізу та перевірки складу продуктів показано на рисунку 8.

Для показника Функціональної адекватності, який визначений формулою (1), значення є оптимальним. Показники Здатності до взаємодії на основі форматів даних, Функціональної перевірки доступу та Ефективності за пропускну здатністю, які визначені формулами (2)-(4) приймають допустимі значення.

Це вказує на належну якість запропонованої інформаційної системи для аналізу та перевірки складу продуктів, а також вказує на можливі шляхи її удосконалення.



**Рис. 8. Значення показників якості інформаційної системи для аналізу та перевірки складу продуктів**

*Джерело: власна розробка*

## 6. Висновки

Отже, запропонована інформаційна система для аналізу та перевірки складу продуктів є невід’ємною частиною сучасного інтегрованого світу, а також важливим та ефективним засобом для безпеки

життя людини. Запропонована структура бази даних словника для відповідної інформаційної системи дозволяє сформувати універсальні таблиці та зв'язки між ними, що дозволить використовувати цю структуру у різних сферах діяльності. Дана робота містить моделювання процесів інформаційної системи для аналізу та перевірки складу продуктів на основі використання CASE-технологій. Зокрема, у роботі містяться зображення взаємодії між акторами та сутностями інформаційної системи для аналізу та перевірки складу продуктів (UML-діаграма використання), структура взаємодії для інформаційної системи (UML-діаграма кооперації), графічне представлення набору дій інформаційної системи (UML-діаграми діяльності), відображення компонент інформаційної системи, залежності та зв'язки між ними (UML-діаграми компонентів), відображення обчислювальних вузлів, компонент, та об'єкти, що виконуються на цих вузлах (UML-діаграма розгортання).

Вибір показників якості інформаційної системи для аналізу та перевірки складу продуктів, що відповідають характеристикам/під-характеристикам якості програмного забезпечення за стандартом ISO/IEC-25010 дають змогу оцінити функціональна адекватність (Functional adequacy) інформаційної системи, її здатність до взаємодії на основі форматів даних (Data exchangeability (Data format based)) та функціональну перевірку доступу (Data exchangeability (User's success attempt based)), а також ефективність за пропускну здатністю (Throughput). Моніторинг показників якості інформаційної системи аналізу та перевірки складу продуктів показав належну якість запропонованої інформаційної системи. А саме, оптимальне значення набув показник Функціональної адекватності, а допустимі – показники Здатності до взаємодії на основі форматів даних, Функціональної перевірки доступу та Ефективності, за пропускну здатністю, що визначає можливі шляхи удосконалення запропонованої інформаційної системи. Результати дослідження використовуються і можуть бути використані для розробки архітектури відповідної інформаційної системи, а визначені показники якості за стандартом ISO/IEC-25010 можуть бути використані для оцінювання якості уже існуючих аналогічних інформаційних систем.

**Список літератури:**

1. Демиденко М. А. Введення в сучасні бази даних : навчальний посібник. НТУ «Дніпровська політехніка». Дніпро : НТУ «ДП», 2020. 38 с.
2. Табунщик Г. В. Проектування та моделювання програмного забезпечення сучасних інформаційних систем : навчальний посібник. Запоріжжя : ЗНТУ, 2016. 270 с.
3. Обод І. та ін. Математичне моделювання інформаційних систем : навчальний посібник. Харківський національний університет радіоелектроніки, 2019. 270 с.
4. Романенко Т. В. Моделювання програмних систем мовою UML. Автоматизація та комп'ютерно-інтегровані технології у виробництві та освіті: стан, досягнення, перспективи розвитку : матеріали Всеукраїнської науково-практичної Internet-конференції. Черкаси, 2021. С. 144–146.
5. Мінухін С. В. Методи і моделі проектування на основі сучасних CASE-засобів : навчальний посібник. Харків : Вид. ХНЕУ, 2008. 272 с
6. Савеленко О. К. CASE-технології у проектуванні інформаційних систем : навч. посіб. Центральноукраїнський національний технічний університет. 2018. 240 с.
7. Погромська Г. С. Оцінка характеристик CASE-засобів на основі однопараметричної моделі Раша. *Вісник КрНУ імені Михайла Остроградського*. 2019. Випуск 5(118). С. 83–89.
8. Погромська Г. С. Раціоналізація визначення об'єктно-орієнтованого CASE-засобу в життєвому циклі програмної системи. *Науковий вісник НЛТУ України: збірник науково-технічних праць*. 2015. Вип. 25.2. С. 356–365.
9. Пелешишин А. М. Методи та алгоритми оптимізації Web-систем. *Вісник Державного університету «Львівська політехніка»*. 2000. № 406. С. 199–211.
10. Шілінг А. Ю. Архітектура он-лайн системи рекомендацій для вдосконалення навчального процесу в закладах вищої освіти. *Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія : Технічні науки*. 2021. Т. 32 (71). Ч. 1. № 2. С. 263–271.
11. Макоєдова В. Проектування інформаційної системи приймальної комісії на основі UML-діаграм. *Наука і техніка сьогодні*. 2023. С. 634–642.
12. Чмуж Г. С. Застосування мови UML для створення розважально-розвиваючих програм. *Наукове мислення : Збірник статей учасників сімнадцятої всеукраїнської практично-пізнавальної конференції «Наукова думка сучасності і майбутнього»*. Дніпро, 2018. С. 42–44.
13. Systems and software engineering. Systems and software Quality-Requirements and Evaluation (SQuaRE). System and software quality models / ISO/IEC 25010.2:2008. URL: [http://sa.inceptum.eu/sites/sa.inceptum.eu/files/Content/ISO\\_25010.pdf](http://sa.inceptum.eu/sites/sa.inceptum.eu/files/Content/ISO_25010.pdf)
14. Software engineering. Product quality. Part 1: Quality model [Electronic resource] / ISO/IEC 9126-1:2001. URL: <http://www.iso.org/iso/en/Catalogue-DetailPage.CatalogueDetail?CSNUMBER=22749&ICS1=35&ICS2=80&ICS3>
15. Software engineering. Product quality.Part 2: External metrics [Electronic resource] / ISO/IEC TR 9126-2:2003. URL: [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_tc/catalogue\\_detail.htm?csnumber](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber)

### References:

1. Demydenko M. A. (2020). Vvedennia v suchasni bazy danykh: navchalnyi posibnyk [Introduction to modern databases: a study guide]. NTU "Dniprovska politehnika". Dnipro: NTU "DP", 38 p. (in Ukrainian)
2. Tabunshchuk H. V. (2016). Proektuvannia ta modeliuvannia prohramnoho zabezpechennia suchasnykh informatsiinykh system: navchalnyi posibnyk [Designing and modeling of software of modern information systems: a study guide]. Zaporizhzhia: ZNTU, 270 p. (in Ukrainian)
3. Obod I. ta in. (2019). Matematyчне modeliuvannia informatsiinykh system: navchalnyi posibnyk [Mathematical modeling of information systems: a study guide]. Kharkivskiy natsionalnyi universytet radioelektroniky, 270 p. (in Ukrainian)
4. Romanenko T. V. (2021). Modeliuvannia prohramnykh system movoiu UML [Modeling of software systems in the UML language]. Avtomatyzatsiia ta kompiuterno-intehrovani tekhnolohii u vyrobnytstvi ta osviti: stan, dosiahnennia, perspektyvy rozvytku: materialy Vseukrainskoi naukovo-praktychnoi Internet-konferentsii. Cherkasy, pp. 144–146. (in Ukrainian)
5. Minukhin S. V. (2008). Metody i modeli proektuvannia na osnovi suchasnykh CASE-zasobiv: navchalnyi posibnyk [Design methods and models based on modern CASE tools: study guide]. Kharkiv: Vyd. KhNEU, 272 p. (in Ukrainian)
6. Savelenko O. K. (2018). CASE-tekhnologii u proektuvanni informatsiinykh system: navchalnyi posibnyk [CASE-technologies in the design of information systems: study guide]. Tsentralnoukrainskyi natsionalnyi tekhnichnyi universytet, 240 p. (in Ukrainian)
7. Pohromska H. S. (2019). Otsinka kharakterystyk CASE-zasobiv na osnovi odnoparametrychnoi modeli Rasha [Evaluation of CASE tools characteristics based on the one-parameter Rasch model]. *Visnyk KrNU imeni Mykhaila Ostrohradskoho*, vol. 5(118), pp. 83–89. (in Ukrainian)
8. Pohromska H. S. (2015). Ratsionalizatsiia vyznachennia obiektno-orientovanoho CASE-zasobu v zhyttievomu tsykli prohramnoi systemy [Rationalization of the definition of an object-oriented CASE tool in the life cycle of a software system]. *Naukovyi visnyk NLTU Ukrainy: zbirnyk naukovo-tekhnichnykh prats*. Lviv: RVV NLTU Ukrainy, vol. 25.2, pp. 356–365. (in Ukrainian)
9. Peleshchyshyn A. M. (2000). Metody ta alhorytmy optymizatsii Web-system [Methods and algorithms for optimizing Web systems]. *Visnyk Derzhavnoho universytetu "Lvivska politehnika"*, no. 406, pp. 199–211. (in Ukrainian)
10. Shilinh A. Yu. (2021). Arkhitektura on-lain systemy rekomendatsii dlia vdoskonalennia navchalnoho protsesu v zakladakh vyshchoi osvity [Architecture of an online recommendation system for improving the educational process in higher education institutions]. *Vcheni zapysky Tavriiskoho natsionalnoho universytetu imeni V. I. Vernadskoho. Seriya: Tekhnichni nauky*, t. 32 (71), ch. 1, no. 2, pp. 263–271. (in Ukrainian)
11. Makoiedova V. (2023). Proiektuvannia informatsiinoi systemy pryimalnoi komisii na osnovi UML-diahram [Designing an information system of the admission committee based on UML diagrams]. *Nauka i tekhnika sohodni*, pp. 634–642. (in Ukrainian)

12. Chmuzh H. S. (2018). Zastosuvannia movy UML dlia stvorennia rozvazhalno-rozvyvaiuchykh prohram [Using the UML language to create entertainment and development programs]. *Naukove myslennia: Zbirnyk statei uchasnykiv simnadtsiatoi vseukrainskoi praktychno-piznavalnoi konferentsii "Naukova dumka suchasnosti i maibutnoho"*. Dnipro, pp. 42–44. (in Ukrainian)

13. Systems and software engineering. Systems and software Quality-Requirements and Evaluation (SQuaRE). System and software quality models / ISO/IEC 25010.2:2008. Available at: [http://sa.inceptum.eu/sites/sa.inceptum.eu/files/Content/ISO\\_25010.pdf](http://sa.inceptum.eu/sites/sa.inceptum.eu/files/Content/ISO_25010.pdf)

14. Software engineering. Product quality. Part 1: Quality model [Electronic resource] / ISO/IEC 9126-1:2001. Available at: <http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=22749&ICS1=35&ICS2=80&ICS3>

15. Software engineering. Product quality. Part 2: External metrics [Electronic resource] / ISO/IEC TR 9126-2:200 3. Available at: [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_tc/catalogue\\_detail.htm?csnumber](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber)