

DOI: <https://doi.org/10.36910/6775-2524-0560-2023-53-36>

УДК 004.715

**Черняшук Наталія Леонідівна**, д.пед.н., професор

<https://orcid.org/0000-0002-3178-8377>

**Бортник Катерина Яківна**, к.т.н., доцент

<http://orcid.org/0000-0001-5282-099X>

**Тишук Максим Анатолійович**, магістрант

**Гнітецький Віталій Віталійович**, магістрант

**Кучер Віталій Леонідович**, магістрант

Луцький національний технічний університет, м. Луцьк, Україна

## МЕТОДИ КЛАСИФІКАЦІЇ АТРИБУТІВ ЯКОСТІ КОМП'ЮТЕРНИХ СИСТЕМ

**Черняшук Н.Л., Бортник К.Я., Тишук М.А., Гнітецький В.В., Кучер В.Л. Методи класифікації атрибутів якості комп'ютерних систем.** В роботі здійснено аналіз наукових праць та практики впровадження методів і засобів для класифікації ознак якості комп'ютера систем, дослідження особливостей процесу забезпечення якості комп'ютерних системи, розробка методу та засобів класифікації ознак якості комп'ютера системи на основі характеристик якості, поданих у текстовій формі. Проведено аналіз предметної області та специфіки процесу класифікації ознак якості комп'ютерних систем, основні сутності та визначено їх властивості, функціональні вимоги до програмного забезпечення. Метод класифікації атрибутів якості комп'ютерних систем. Це дозволило проектувати та реалізовувати архітектуру системи з урахуванням особливостей Метод QFD та запропонована процедура класифікації з використанням мови C# та Технології ASP NET MVC.

**Ключові слова:** метод, засіб, класифікація, атрибут, комп'ютерна система, якість.

**Chernyashchuk N., Bortnyk K.Y., Tyschuk M., Hnitetyskiy V., Kucher V. Methods of classification of quality attributes of computer systems.** The main tasks of the master's thesis are the analysis of scientific works and the practice of implementing methods and tools for the classification of computer system quality features, researching the features of the process of ensuring the quality of computer systems, developing a method and means of classifying computer system quality features based on the quality characteristics presented in the text form. An analysis of the subject area and the specifics of the process was carried out, the classification of the quality features of computer systems, the main entities and their properties were determined, the functional requirements for the software. The method of classification of the quality attributes of computer systems. classification using the C# language and ASP NET MVC technology.

**Key words:** method, tool, classification, attribute, computer system, quality.

**Постановка наукової проблеми.** Визначення атрибутів якості комп'ютерних систем, важливий крок у розробці та управлінні програмним забезпеченням. Існує кілька алгоритмів та методів, які можуть бути використані для цього завдання. Однак конкретний алгоритм може варіюватися в залежності від контексту та вимог конкретного проекту.

Для визначення основних зацікавлених сторін (стейкхолдерів), які взаємодіють з системою. Ними можуть бути користувачі, клієнти, менеджери, розробники, тестувальники тощо. Потрібно провести спільні обговорення та інтерв'ю для збору вимог від стейкхолдерів. Визначити, які аспекти системи є для них критичними з точки зору якості. Виокремити та класифікувати атрибути якості відповідно до міжнародних стандартів, таких як ISO/IEC 25010. Типові категорії включають продуктивність, ефективність, надійність, сумісність, підтримку і т. д. Встановити конкретні метрики для вимірювання кожного атрибуту якості. Наприклад, для продуктивності це може бути час відповіді, для надійності - кількість відмов тощо. Визначити, які значення метрик вважатимуться прийнятними та неприйнятними для кожного атрибуту якості. Зафіксувати всі визначені атрибути якості, метрики та порогові значення у вимогах до системи. Це може бути представлено у вигляді специфікації вимог чи іншого документа. Здійснити аналіз і оцінку визначених атрибутів якості для визначення, наскільки система відповідає вимогам. Врахувати, що визначення атрибутів якості - це ітеративний процес. З урахуванням змін у вимогах чи умовах використання, оновлюйте атрибути якості відповідно.

**Аналіз досліджень.** Даний підхід може бути адаптований в залежності від конкретних потреб та контексту проекту. Важливо взаємодіяти зі стейкхолдерами та забезпечити, що визначені атрибути якості відображають реальні потреби користувачів та вимоги бізнесу.

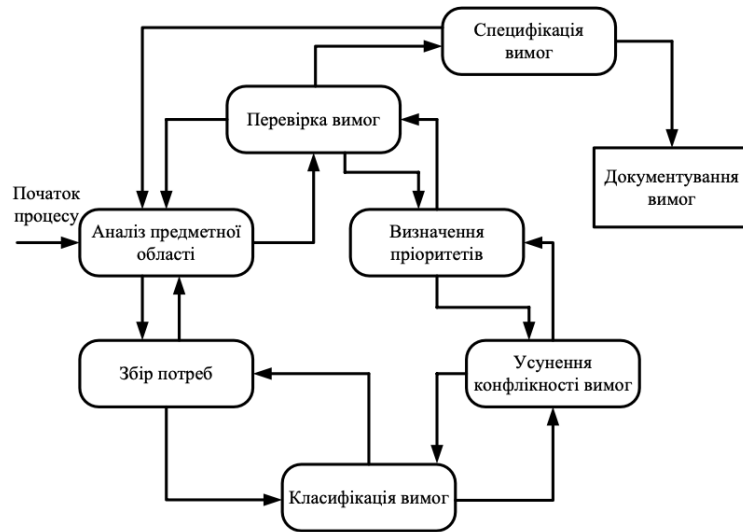


Рис. 1. Алгоритм розробки і аналізу КС [2]

При використанні даного підходу на основі стандартних моделей якості ISO/IEC 25010 для формування вимог, схема процесу їх розробки та аналізу.

Пропонується перетворити з форми, зображеної на рисунку 1, наприклад показано на рисунку 2.

З аналізу діаграми алгоритму процесу розробки та аналізу вимог, наведених в рисунку 2, видно, що формування вимог здійснюється за трьома моделями якості, які дозволяють структурувати і забезпечувати їх спілкування одночасно стандартизація.

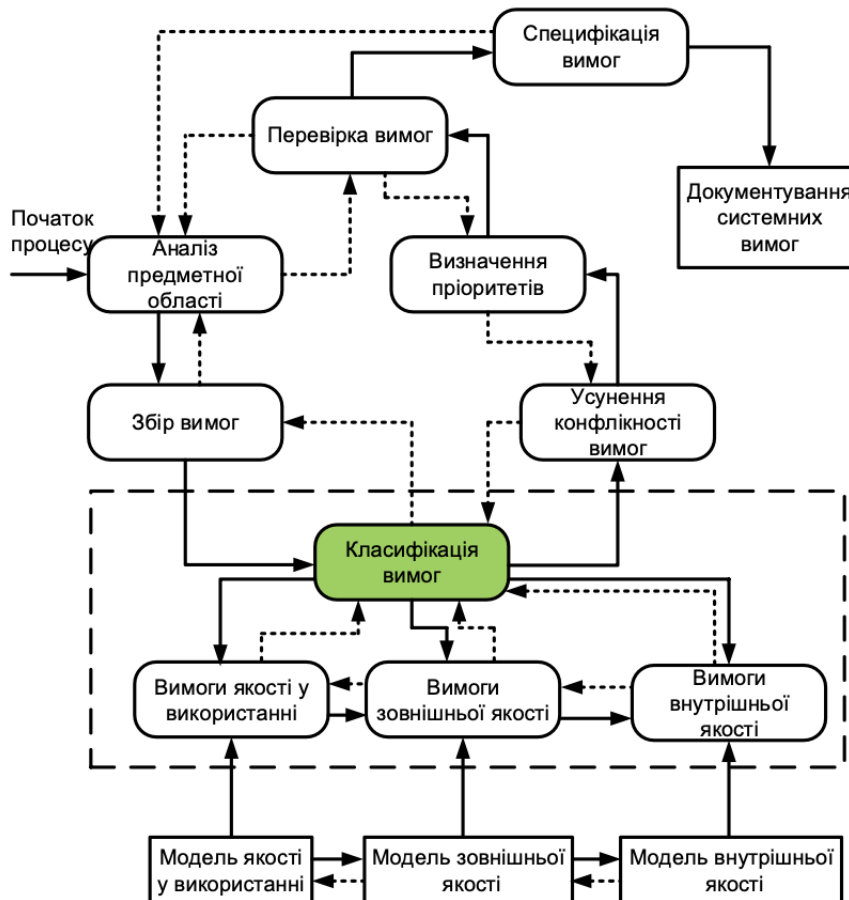


Рис. 2. Алгоритм розробки і аналізу КС на основі МЯ [4]

Автоматизувати процес класифікації ознак за ознаками якості, необхідно визначити характеристики для кожної характеристики якості в трьох моделі: модель якості у використанні, зовнішня модель і внутрішня модель якості.

**Мета роботи.** Полягає у дослідженні методів і засобів класифікації атрибутів якості комп'ютерних систем, представлених у текстовому вигляді, за стандартизованими характеристиками якості для підвищення ефективності реалізації комп'ютерних систем.

**Виклад основного матеріалу й обґрунтування отриманих результатів дослідження.** Визначення ознак характеристик якості та обґрунтування формалізованого представлення моделей якості важливо для ефективного управління якістю комп'ютерних систем. Ознаки характеристик якості:

**Продуктивність:** швидкодія, відгук, завантаження системи. Приклади метрик – час відповіді, пропускна здатність, завантаження сервера.

**Ефективність:** використання ресурсів, оптимізація алгоритмів. Приклади метрик – витрати пам'яті, час виконання алгоритму.

**Надійність:** стабільність, відновлюваність, здатність до відновлення. Приклади метрик – кількість відмов, середній час між відмовами.

**Сумісність:** взаємодія з іншими системами, сумісність інтерфейсів. Приклади метрик – рівень сумісності зі стандартами, кількість інтерфейсних конфліктів.

**Безпека:** керування доступом, захист від атак, конфіденційність даних. Приклади метрик – кількість виявлених уразливостей, час виявлення та реагування на інциденти безпеки.

**Тривалість життя та обслуговування:** підтримка, легкість модифікації, масштабованість. Приклади метрик – час відновлення після виправлення помилок, час налагодження нового функціоналу.

Формалізоване представлення моделей якості:

ISO/IEC 25010 – міжнародний стандарт, який надає модель для характеристик якості та підхарактеристик. Характеристики представлені як високорівневі категорії, а підхарактеристики визначають конкретні аспекти цих категорій.

Модель FURPS+:

Визначає п'ять категорій атрибутів якості: функціональність, надійність, витрати, продуктивність та сумісність. Кожна категорія визначається через конкретні атрибути та їхні вимоги.

Модель Боема:

Визначає п'ять основних атрибутів якості: продуктивність, надійність, зручність використання, технічна підтримка та сумісність. Кожен атрибут визначається через конкретні підхарактеристики та метрики.

Формалізоване представлення моделей якості може включати математичні вирази, графіки, табличні представлення та інші засоби для чіткого визначення та вимірювання атрибутів якості. Основна ідея полягає в тому, щоб забезпечити точне і формалізоване визначення критеріїв, які визначають якість системи.

Вимоги зовнішньої якості (external quality requirements) програмного забезпечення (ПЗ) визначають, як система повинна вести себе в операційному середовищі та як користувачі взаємодіють із системою. Ці вимоги визначають характеристики, які можна спостерігати під час роботи системи. Декілька типових вимог зовнішньої якості включають [2]:

**Ефективність** – система повинна реагувати на користувацькі запити протягом прийняттого часу.

**Продуктивність** – система повинна забезпечувати ефективне використання ресурсів.

**Стабільність** – система повинна залишатися стабільною під час роботи без частих відмов або збоїв.

**Відновлення** – система повинна відновлюватися після відмов або збоїв без втрати даних або важливої інформації.

**Взаємодія з іншими системами** – система повинна ефективно взаємодіяти з іншими системами, які використовуються в організації.

**Сумісність з ПЗ** – система повинна бути сумісною з різними операційними системами та програмним забезпеченням.

**Інтерфейс користувача** – інтерфейс повинен бути зрозумілим, зручним та ергономічним.

Надається достатня документація, яка допомагає користувачам ефективно використовувати систему.

Захист даних – забезпечується захист конфіденційності та цілісності даних.

Керування доступом – визначається система управління доступом для запобігання несанкціонованому доступу.

Легкість модифікації – система повинна бути легкою для модифікацій та оновлень.

Технічна підтримка – забезпечується належна технічна підтримка та обслуговування.

Переносимість – система повинна бути портативною між різними платформами та середовищами.

Ці вимоги визначаються з урахуванням потреб користувачів, бізнес-вимог, а також урахуванням можливостей і обмежень технічної архітектури та інфраструктури. Якість зовнішньої взаємодії та характеристики роботи системи є важливими елементами для задоволення потреб користувачів і успіху проекту [5].



Рис. 3. Алгоритм зовнішньої якості

Формалізація моделі внутрішньої якості програмного забезпечення включає визначення ієрархічної структури залежностей між характеристиками внутрішньої якості, аналогічно до моделі зовнішньої якості. Основною метою внутрішньої якості є забезпечення ефективності, обслуговуваності та розширюваності системи. Загальні категорії та ієрархічна структура для внутрішньої якості [4]:

Структура та Архітектура – розділення системи на логічні модулі для спрощення розробки та обслуговування; ефективне управління та збереження даних в системі.

Ефективність та Оптимізація – ефективне використання пам'яті та обчислювальних ресурсів; мінімізація часу виконання та оптимізація алгоритмів.

Безпека та Надійність – захист системи від зловмисників та забезпечення цілісності даних; механізми для виявлення та виправлення помилок.

Тестування та Валідація – визначення методів та критеріїв для внутрішнього тестування; перевірка правильності та валідності введених даних.

Обслуговуваність та Легкість Модифікації – можливість додавання нового функціоналу без зміни вже існуючого коду; забезпечення легкості управління та виправлення системи.

Документація та Коментарі – наявність докладної та зрозумілої документації до коду; забезпечення належної документації через коментарі в коді.

Використання Стандартів та Інструментів – використання стандартів розробки для забезпечення якості коду; використання ефективних інструментів для розробки, тестування та валідації.

Формалізація цих характеристик включає визначення метрик, критеріїв, правил та стандартів, які дозволяють вимірювати та оцінювати різні аспекти внутрішньої якості програмного забезпечення. Такий підхід визначає, наскільки добре система відповідає стандартам та вимогам внутрішньої якості [3, 4].

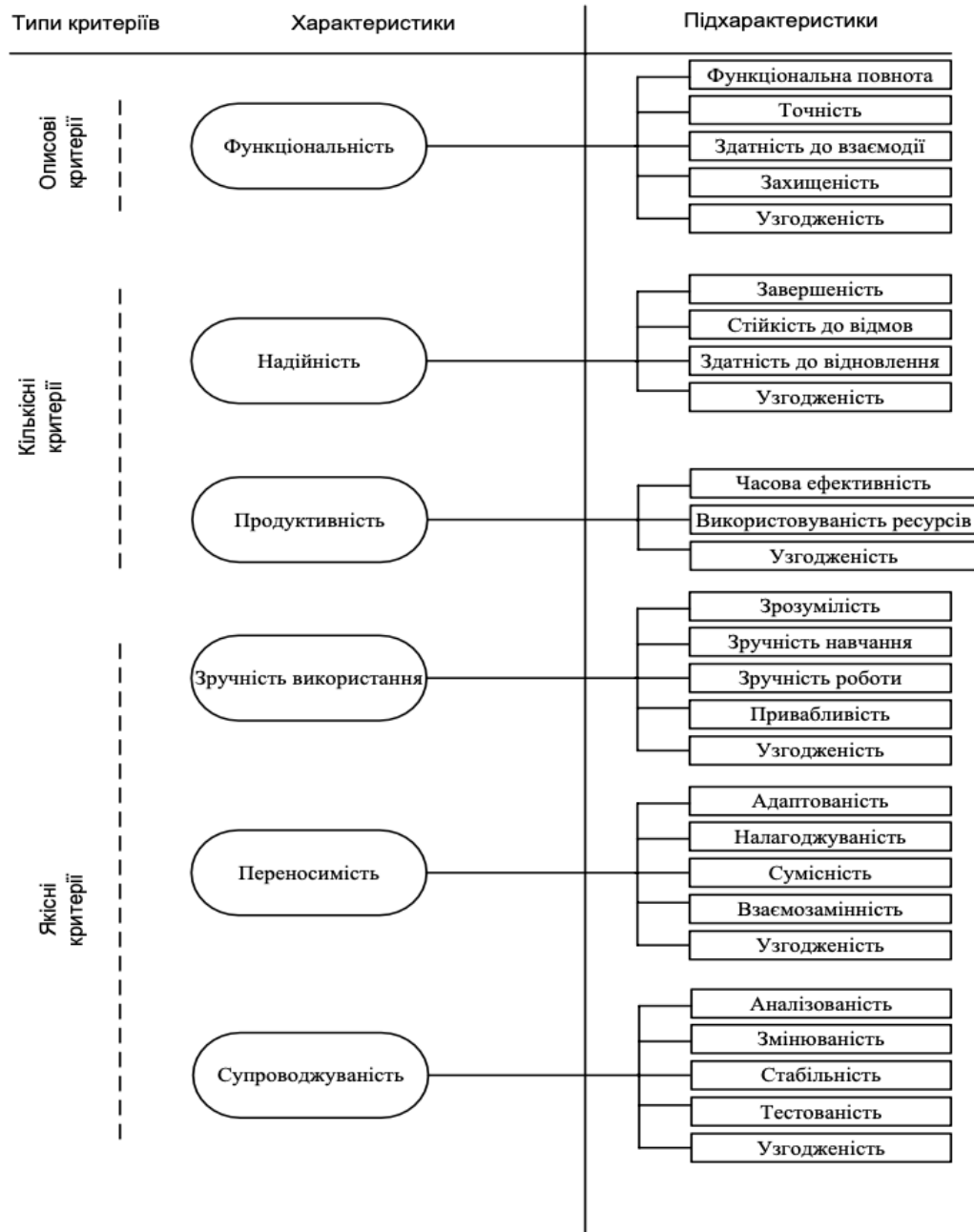


Рис. 4. Алгоритм ВЯ

**Висновки.** Проведено аналіз успішності проектів комп'ютерних систем, у тому числі програмних компонентів, та визначено негативні фактори впливу як кінцевої системи. Основні з них – неточність або неадекватність. Класифікація атрибутів та показників якості відповідно до стандартизованих характеристик потребує додаткового дослідження методів та засобів класифікації, зокрема тексту даних.

Проведено аналіз моделей якості для оцінки відповідності заявлених вимог до комп'ютерної системи реалізованим властивостям, що дозволило обґрунтувати використання моделей стандарту ISO/IEC 25010 для представлення характеристик комп'ютерної системи.

Проаналізовано принципи та структуру процесу класифікації текстової інформації, що дозволило виявити потенційні інтелектуальні методи класифікації ознак якості комп'ютерних систем за стандартизованими наборами характеристик.

Обґрунтовано алгоритм класифікації ознак якості комп'ютера. системи, яка полягає у використанні моделей якості стандарту ISO/IEC 25010 для представлення вимог до комп'ютерних систем і дає змогу визначити класи, на які класифікуються вимоги та реалізовані властивості комп'ютерної системи.

#### Список бібліографічного опису

1. Мар'ян М. В. Алгоритми ідентифікації об'єктів Інтернет речей на основі технології блокчейн / Algorithms for Internet things objects identifying based on blockchain technology [Електронний ресурс] / МУРИН Васильович Мар'ян // dspace.wunu.edu.ua. – 2018. – Режим доступу до ресурсу: [http://dspace.wunu.edu.ua/jspui/bitstream/316497/32438/1/%D0%9A%D1%96%D0%BC21\\_%D0%9C%D1%83%D1%80%D0%B8%D0%BD\\_%D0%9C.pdf](http://dspace.wunu.edu.ua/jspui/bitstream/316497/32438/1/%D0%9A%D1%96%D0%BC21_%D0%9C%D1%83%D1%80%D0%B8%D0%BD_%D0%9C.pdf).
2. Brooks T. The key to understanding wireless connectivity in the industrial IoT [Електронний ресурс] / Tim Brooks // Infovista. – 2020. – Режим доступу до ресурсу: <https://www.infovista.com/blog/understanding-wireless-connectivity-in-the-industrial-iot>.
3. Heiney S. Industrial Internet of Things (IIoT) Security: Everything You Need to Know [Електронний ресурс] / Sam Heiney // ImperoSoftware. – 2021. – Режим доступу до ресурсу: <https://www.imperosoftware.com/industrial-internet-of-things-complete-guide/>.
4. Shea S. LPWAN (low-power wide area network) [Електронний ресурс] / Sharon Shea // IoT Agenda. – 2017. – Режим доступу до ресурсу: <https://internetofthingsagenda.techtarget.com/definition/LPWAN-low-power-wide-area-network>.
5. EMnify. Cellular IoT: What Business Leaders Should Know [Електронний ресурс] / EMnify // EMnify. – 2020. – Режим доступу до ресурсу: <https://www.emnify.com/blog/cellular-iot>.
6. Acosta G. The ZigBee Protocol [Електронний ресурс] / Gonzalo Acosta // netguru. – 2018. – Режим доступу до ресурсу: <https://www.netguru.com/blog/the-zigbee-protocol>.
7. Digi I. Understanding the Zigbee 3.0 Protocol [Електронний ресурс] / International Digi // digi. – 2018. – Режим доступу до ресурсу: <https://www.digi.com/blog/post/understanding-the-zigbee-3-0-protocol>.
8. Wireless mesh network [Електронний ресурс] // Wikipedia – Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/Wireless\\_mesh\\_network](https://en.wikipedia.org/wiki/Wireless_mesh_network).
9. Почему Bluetooth IoT? [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://www.mokoblue.com/ru/why-bluetooth-iot/>.
10. Вичугова А. RFID [Електронний ресурс] / Анна Вичугова // bigdataschool. – 2019. – Режим доступу до ресурсу: <https://www.bigdataschool.ru/wiki/rfid>.
11. Weiss J. Wireless Sensor Networking for the Industrial Internet of Things [Електронний ресурс] / J. Weiss, Y. Ross // analog – Режим доступу до ресурсу: <https://www.analog.com/ru/technical-articles/wireless-sensor-networking-for-ind-iiot.html>.
12. Sisinni E. Wireless Communications for Industrial Internet of Things: The LPWAN Solutions [Електронний ресурс] / E. Sisinni, A. Mahmood // ResearchGate. – 2021. – Режим доступу до ресурсу: [https://www.researchgate.net/publication/347587880\\_Wireless\\_Communications\\_for\\_Industrial\\_Internet\\_of\\_Things\\_The\\_LPWAN\\_Solutions](https://www.researchgate.net/publication/347587880_Wireless_Communications_for_Industrial_Internet_of_Things_The_LPWAN_Solutions).

#### References

1. Maryan M. V. Algorithms for Internet of things identifying objects based on blockchain technology / Algorithms for Internet things objects identifying based on blockchain technology [Electronic resource] / Maryan MURYAN // dspace.wunu.edu.ua. – 2018. – Resource access mode: [http://dspace.wunu.edu.ua/jspui/bitstream/316497/32438/1/%D0%9A%D1%96%D0%BC21\\_%D0%9C%D1%83%D1%80%D0%B8%D0%BD\\_%D0%9C.pdf](http://dspace.wunu.edu.ua/jspui/bitstream/316497/32438/1/%D0%9A%D1%96%D0%BC21_%D0%9C%D1%83%D1%80%D0%B8%D0%BD_%D0%9C.pdf).
2. Brooks T. The key to understanding wireless connectivity in the industrial IoT [Electronic resource] / Tim Brooks // Infovista. – 2020. – Mode of access to the resource: <https://www.infovista.com/blog/understanding-wireless-connectivity-in-the-industrial-iot>.
3. Heiney S. Industrial Internet of Things (IIoT) Security: Everything You Need to Know [Electronic resource] / Sam Heiney // ImperoSoftware. – 2021. – Mode of access to resources at: <https://www.imperosoftware.com/industrial-internet-of-things-complete-guide/>.
4. Shea S. LPWAN (low-power wide area network) [Electronic resource] / Sharon Shea // IoT Agenda. – 2017. – Resource access mode: <https://internetofthingsagenda.techtarget.com/definition/LPWAN-low-power-wide-area-network>.
5. EMnify. Cellular IoT: What Business Leaders Should Know [Electronic resource] / EMnify // EMnify. – 2020. – Resource access mode: <https://www.emnify.com/blog/cellular-iot>.
6. Acosta G. The ZigBee Protocol [Electronic resource] / Gonzalo Acosta // netguru. – 2018. – Resource access mode: <https://www.netguru.com/blog/the-zigbee-protocol>.
7. Digi I. Understanding the Zigbee 3.0 Protocol [Electronic resource] / International Digi // digi. – 2018. – Resource access mode: <https://www.digi.com/blog/post/understanding-the-zigbee-3-0-protocol>.

8. Wireless mesh network [Electronic resource] // Wikipedia – Resource access mode: [https://en.wikipedia.org/wiki/Wireless\\_mesh\\_network](https://en.wikipedia.org/wiki/Wireless_mesh_network).
9. Why Bluetooth IoT? [Electronic resource]. – 2020. – Mode of access to the resource: <https://www.mokoblue.com/ru/why-bluetooth-iot/>.
10. Vychugova A. RFID [Electronic resource] / Anna Vychugova // bigdataschool. – 2019. – Mode of access to resource: <https://www.bigdataschool.ru/wiki/rfid>.
11. Weiss J. Wireless Sensor Networking for the Industrial Internet of Things [Electronic resource] / J. Weiss, Y. Ross // analog – Resource access mode: <https://www.analog.com/ru/technical-articles/wireless-sensor-networking-for-ind-iiot.html>.
12. Sisinni E. Wireless Communications for Industrial Internet of Things: The LPWAN Solutions [Electronic resource] / E. Sisinni, A. Mahmood // ResearchGate. – 2021. – Mode of access to resources at: [https://www.researchgate.net/publication/347587880\\_Wireless\\_Communications\\_for\\_Industrial\\_Internet\\_of\\_Things\\_The\\_LPWAN\\_Solutions](https://www.researchgate.net/publication/347587880_Wireless_Communications_for_Industrial_Internet_of_Things_The_LPWAN_Solutions).