

Міністерство освіти та науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій і технічного захисту інформації
(повна назва)

Кафедра Радіотехнологій інформаційно-комунікаційних систем
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

рівень вищої освіти перший (бакалавр)

ІНФОРМАТИЗАЦІЯ ПІДПРИЄМСТВА. ПРОГРАМНО- АПАРАТНИЙ ПРИСТРІЙ УРАХУВАННЯ ЧАСУ ПРОСТОЮ ПРАЦІВНИКА

(тема)

Виконав:

студент IV курсу, групи ІТІР-20-1
Скрипник К.О.
(прізвище, ініціали)

Спеціальність 126 Інформаційні системи та технології
(код і повна назва спеціальності)

Освітня програма Інформаційні технології інтернету речей
(повна назва освітньої програми)

Керівник ст. викл Алфьоров М.Є.
(посада, прізвище, ініціали)

Допускається до захисту

В.о. зав. кафедри

(підпис)

Зарудний О.А.
(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій і технічного захисту інформації

Кафедра Радіотехнологій інформаційно-комунікаційних систем

Рівень вищої освіти перший (бакалаврський)

Спеціальність 126 Інформаційні системи та технології

(код і повна назва)

Тип програми Освітньо-професійна

Освітня програма Інформаційні технології інтернету речей

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«_____» _____ 2024 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Скрипник Крестині Олександрівні

(прізвище, ім'я, по батькові)

1. Тема роботи Інформатизація підприємства. Програмно-апаратний пристрій урахування часу простою працівника

затверджена наказом по університету від 27.05.2024 . №500Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10.06.2024

3. Вихідні дані до роботи

3.1 Провести огляд та аналіз аналогічних систем

3.2 Розробити структурну схему пристрою

3.3 Обрати елементну базу для реалізації пристрою

3.4 Розробити програмне забезпечення роботи пристрою

4. Перелік питань, що потрібно опрацювати в роботі 1 ВИКОРИСТАННЯ RFID ТЕХНОЛОГІЙ У ПОВСЯКДЕННІ

Вступ. 1 Огляд та аналіз аналогічних систем. 2 Розробка структурної схеми пристрою. 3 Вибір та обґрунтування елементної бази. 4 Розробка програмного забезпечення. Висновки. Перелік джерел посилання. Додатки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій_(слайдів) Комп'ютерна презентація

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Обговорення теми	06.05.2024	Вик.
2	Отримання завдання	10.05.2024	Вик.
3	Розробка першого розділу	14.05.2024	Вик.
4	Розробка другого розділу	19.05.2024	Вик.
5	Написання коду	28.05.2024	Вик.
6	Обговорення деталей	30.05.2024	Вик.
7	Перевірка роботи коду	02.06.2024	Вик.
8	Завершення роботи	10.06.2024	Вик.
9	Захист	21.06.2024	

Дата видачі завдання 06.05.2024

Студент _____
(підпис)

Скрипник К.О.

Керівник роботи _____
(підпис)

Алфьоров М.Є.

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи бакалавра містить 77 сторінок тексту, 33 рисунків, 12 Таблиць, 11 джерел посилання і 3 додатка.

LCD-ЕКРАН. ДАТЧИК. ЗЧИТУВАННЯ ІНФОРМАЦІЇ.

ЧІП. КАРТКА. ЧАС. ПРИСТРІЙ.

Об'єктом розробки є пристрій - автоматизованого зчитування особистої інформації з робочих карток працівників.

Метод дослідження – описово-аналітичний

Метою даної кваліфікаційної дипломної роботи бакалавра є схематична та програмна розробка автоматизованої системи пристрою зчитування та відстеження інформації кожного працівника.

В даній роботі проведено огляд і аналіз вимог до правильної роботи екрану, огляд і аналіз систем автоматичного зчитування інформації, розроблено структурну схему системи, обрано елементну базу для реалізації електронного екрану системи та запропоновано програмне забезпечення управління системою.

ABSTRACT

The qualification work of the bachelor consists of an explanatory note containing 77 pages of text, 33 figures, 12 tables, 11 literary sources and 3 appendices.

LCD SCREEN. SENSOR. READING INFORMATION.

CHIP CARD. TIME. DEVICE.

The object of development is a device for automated reading of personal information from employee work cards

Research method - descriptive and analytical

The purpose of this bachelor's qualification work is the schematic and software development of an automated system and a device for reading and scanning the information of each employee.

In this work, a review and analysis of the requirements for the correct operation of the screen, a review and analysis of automatic information reading systems, a structural diagram of the system was developed, an element base was selected for the implementation of the electronic screen of the system, and the system management software was proposed.

ЗМІСТ

Вступ.....	8
1 ПРОБЛЕМИ ТА ЗАСТОСУВАННЯ ТЕХНОЛОГІЇ RFID ДЛЯ ЗБОРУ ДАНИХ	10
1.1 Автоматизовані системи обліку енергоресурсів.....	10
1.2 Завдання збору даних про прилади обліку	11
1.3 Огляд технології RFID.....	12
1.4 Огляд застосування технології RFID	16
1.5 Порівняльний аналіз радіочастотної ідентифікації з іншими	20
Технології.....	Error! Bookmark not defined.
ВИСНОВКИ ДО РОЗДІЛУ 1	22
2 РОЗРОБКА ОНТОЛОГІЧНОЇ МОДЕЛІ СИСТЕМИ ЗБОРУ ДАНИХ ПРО ЕНЕРГОСПОЖИВАННЯ	24
2.1 Використання онтології для дослідження предметної області	24
2.2 Побудова метаонтології програмних засобів.....	25
2.3 Побудова онтології програмних засобів.....	29
2.4 Розробка методів верифікації та валідації для аналізу даних	33
Облік енергії	33
ВИСНОВКИ ДО РОЗДІЛУ 2	35
3 РОЗРОБКА ТА ДОСЛІДЖЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	35
3.1 Вибір типу зчитувача та міток для збору даних лічильників.....	Error! Bookmark not defined.
3.2 Оцінка трудомісткості	Error! Bookmark not defined.
3.3 Аналіз вимог до архітектури системи.....	Error! Bookmark not defined.
3.4 Аналіз вимог	36
3.5 Розробка програмного забезпечення.....	40

	7
3.6 Тестування програмного забезпечення.....	46
3.7 Аналіз якості коду	50
3.8 Експериментальне дослідження програмних засобів	5251
ВИСНОВКИ ДО РОЗДІЛУ 3	Error! Bookmark not defined.
ВИСНОВОК.....	55
Список літератури.....	55
Додаток А. РЕЗУЛЬТАТИ ТЕСТУВАННЯ	57
Додаток Б. СЛАЙДИ ПРЕЗЕНТАЦІЇ	70
Додаток В. ВІДОМОСТІ АТЕСТАЦІЙНОГО ПРОЕКТУ	
76	

ВСТУП

Назва навчальної програми: «Інформатизація підприємства. Програмно-апаратний пристрій урахування часу простою працівника».

Актуальність теми роботи

У наш час RFID-системи увійшли в життя людей і використовуються в наступних сферах: логістика, торгівля, контроль доступу, виробництво, розподіл і системи управління бухгалтерським обліком. Технологія RFID дозволяє уникнути ручного збору даних і не вимагає контакту мітки зі зчитувачем. Дана технологія дозволяє заощадити багато часу, так як при великому обсязі даних час, пов'язаний з ручним збором інформації, може бути досить значним.

В автоматизованих системах енергоменеджменту та моніторингу використання технології RFID дозволяє вирішити, наприклад, такі завдання: початкове формування інформації про прилади обліку, налаштування автоматизованої системи, автоматизована зміна даних, тощо.

Використання технології радіочастотної ідентифікації дозволяє значно знизити витрати на створення та проектування систем обліку енергії. Це значно скорочує час, витрачений на збір даних і налаштування системи.

Метою роботи є розробка програмного засобу та пристрою збору даних працівників, для визначення часу простою на підприємстві за допомогою технології радіочастотної ідентифікації.

Об'єктом дослідження дипломної роботи є програмне забезпечення та прилад для виводу даних з картки на екранчик.

Тематика роботи – технологія радіочастотної ідентифікації.

Для досягнення поставленої мети в ході дослідження повинні бути вирішені наступні наукові завдання :

- аналіз існуючих систем обліку для збору даних про всіх працівників;

- розробка онтологічної моделі програмних засобів збору даних про облік з використанням технології радіочастотної ідентифікації;
- розробка методів верифікації даних, отриманих за допомогою RFID-технології;
- Розробка програмних засобів збору даних та розробка LCD-пристрою;
- оцінка ефективності розробленого девайсу.

Для рішення задач дисертаційної роботи були використані наступні методи дослідження й розробки :

- онтологічний аналіз;
- об'єктно-орієнтований підхід до створення програмного девайсу;
- системно-інформаційний аналіз й менеджмент якості.

Научна тематика дипломної роботи

- розроблена онтологічна модель програмних виробів збору даних про облік інформації робітників на простою з використанням технології радіочастотної ідентифікації;
- розроблено архітектуру програмних засобів збору даних про простій робітників з використанням LCD-екранчику;
- розроблено програмні засоби збирання даних про час простою працівника за допомогою особистої картки користувача з чіпом
- розроблено програмні засоби вичислення інформації з використанням RFID технологій.

Практична значимість дослідження

Було створено програмні засоби та реалізовані методи, які дозволяють суттєво зменшити помилки щодо відсутності робітників на робочих місцях.

Публікації. По матеріалам дисертації було опубліковано мінімум 12 статей.

Структура роботи та об'єм роботи. Робота написана на 83 листах машинописного тексту та складається з введення, трьох заголовків, висновку, списку літератури, який має 11 істочників, чотирьох додатків тощо.

1 ЗАДАЧА ТА ОБЛАСТІ ВИКОРИСТАННЯ RFID ТЕХНОЛОГІЙ ДЛЯ СБОРУ ДАНИХ

1.1 Автоматизація системи обліку

Одним із головних завдань у розвитку технологічних підприємств є зростання рівня технічної означеності інженерних об'єктів та використання сучасних інформаційних технологій для контролю роботи. Насамперед, це робітники, які займаються забезпеченням енергоресурсів – датчиками, контроллерами та екранами зчитування інформації, підвищувальними станціями та об'єктами споживання енергоресурсів – виробничі, житлові та адміністративні підприємства для створення гаджетів.

Автоматизація та диспетчеризація переслідує кілька цілей:

- комфорт користувачів;
- цілісність інформації;
- технологічна безпека;
- зниження витрат на експлуатацію;
- комерційний облік використання ресурсів.

Структура систем автоматизації зазвичай включає кілька рівнів. Нижнім рівнем будь-яких систем диспетчеризації є контрольно-вимірювальна апаратура, пристрої автоматизованого керування.

Через канали зв'язку, які можуть мати різну природу, здійснюється з'єднання пристроїв автоматики та операторських станцій диспетчеризації. Тенденція сьогоdnішнього дня – відмова від пристроїв локального регулювання із заздалегідь заданою логікою роботи та означення об'єктів універсальними програмованими контролерами.

Пристрої автоматики повинні об'єднуватись між собою лініями передачі цифрових даних, що створює єдиний інформаційний простір. Для системи автоматизації та диспетчеризації, в якому знаходяться підсистеми

зчитування інформації, вивід даних, які мають різну природу та сенс у використанні.

На рисунку 1.1 приведена структура системи автоматизованого зчитування даних з картки з чіпом та виведення на LCD екран.

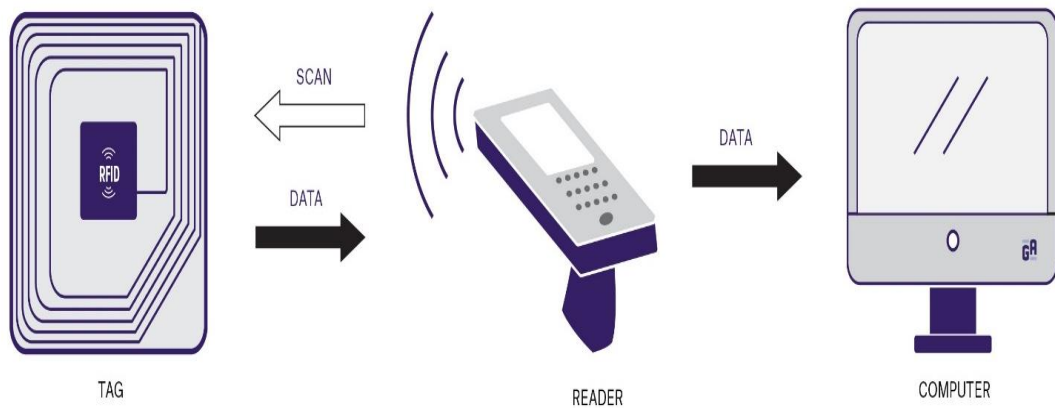


Рисунок 1.1 – Структура системи автоматизованого датчику

Універсальність забезпечується підтримкою широкого набору стандартних типів сигналів та інтерфейсів для підключення обладнання – виконавчих механізмів та вимірювальних датчиків.

На базі програмованих контролерів, згідно з довільним технічним завданням, інженер може створювати системи управління під кожен індивідуальний об'єкт, враховуючи всі його особливості та додаткові вимоги замовника.

Завдяки легкому використанню та зручності, кожен працівник без проблем може відстежити свій час перерви та не запізнитися на робоче місце. Розробка даного приладу допоможе правильно використовувати час та зменшити витрати, тим самим заробляти більше грошей та надавати якісніші послуги.

1.2 Задачі збору інформації о пристроях обліку

Під час експлуатації систем виникає завдання збору даних про прилади обліку. Ручний збір інформації є дуже трудомістким завданням, забирає багато

часу, не забезпечує необхідний рівень достовірності даних, оскільки зібрані дані практично не піддаються належної та якісної перевірки.

Також, існують наступні задачі:

- Початкове формування відомостей про прилади обліку після їх встановлення;
- Заміна даних та перефігурування системи при додаванні/заміні/видаленні приладів обліку;
- збереження даних про зміни характеристик та показань приладу у сховищі.

Вирішення цих завдань особливо актуальне на етапах розгортання системи та її сервісного обслуговування під час експлуатації автоматизованих систем на будь-якому підприємстві.

1.3 Вигляд технології радіочастотної ідентифікації

Радіочастотна ідентифікація (RFID - Radio Frequency Identification) - метод автоматичної ідентифікації об'єктів, в якому для зчитування та запису інформації використовуються радіосигнали.

Антенна зчитувача посилає сигнал до мітки, потім інтегральна схема всередині мітки виявляє сигнал зчитувача, після чого відбувається зворотна передача даних. Принцип дії технології RFID показаний рисунку 1.2.

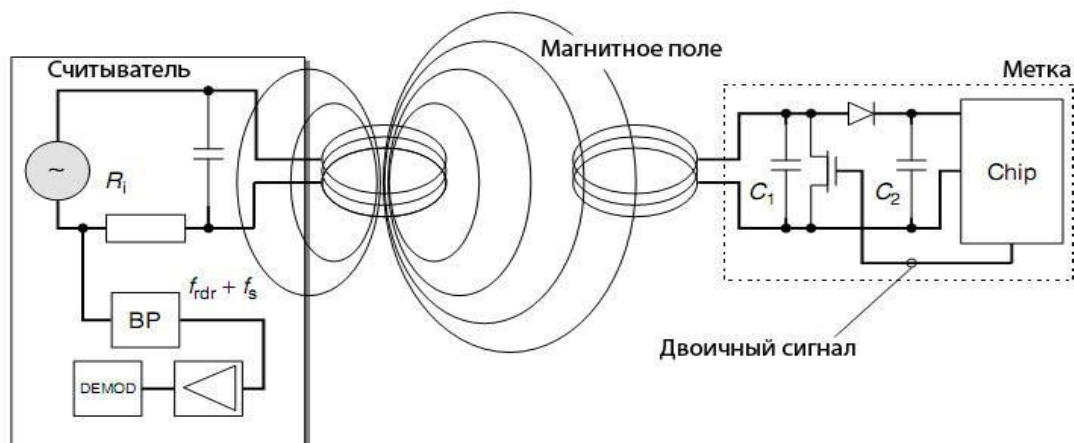


Рисунок 1.2 – Принцип дії технології RFID

Вважають, що RFID-технологія почала з'являтися під час Другої світової війни. Американські, японські, британські та німецькі фахівці використали радар, розроблений у 1935 році фізиком із Шотландії, Робертом Олександром Вотсоном-Уоттом, для попередження наземних сил про наближення літаків.

Тоді існувала проблема – був спосіб ідентифікації приналежності літаків: не можна було розрізнити одиниці техніки ворожих і дружніх військ. Німецькі фахівці виявили одну цікаву залежність, коли пілоти при наближенні до бази здійснювали певний маневр - подвійний переворот через крило, то радіосигнал, що відбивався від крила, змінювався.

Цей метод повідомляв війська на суші, що це наближалися німецькі літаки, а не техніка інших країн. Це використання радіохвиль для ідентифікації об'єктів було прототипом першої пасивної RFID-системи.

У рамках секретного проекту, який очолює Уотсон-Уотт, британські вчені створили першу активну систему для розпізнавання об'єктів, яка працювала за принципом «свій-чужий».

На британських літаках розташовувався передавач. Після отримання літаком сигналу від наземної станції відбувалася передача зворотного сигналу. Факт отримання сигналу наземною службою означав, що літак є дружнім. Той самий основний принцип використовується і при повсякденному застосуванні цієї технології. Сигнал передається радіомітці, яка відображає отриманий сигнал (у разі пасивної системи) або сам транслює сигнал (у випадку активної системи)

Використання радіочастотної ідентифікації передачі даних і виявлення об'єктів тривало у 50-ті і 60-ті роки 20-го століття. Вчені та фахівці з різних країн проводили дослідження та подавали звіти, в яких було роз'яснено, яким чином дана технологія могла бути впроваджена у системи для ідентифікації об'єктів на відстані.

Компанії почали активно розробляти та розвивати протикрадіжні системи, що приносило величезний дохід. Дані системи використовували радіохвилі визначення статусу оплати товару у магазині. Протикрадіжні радіочастотні мітки досі активно використовуються. Об'єм їх пам'яті становить 1 біт. Можливі 2 стани: біт активний, біт не активний.

Після оплати товару покупцем мітка відключається, після чого покупець може залишити магазин. У разі відсутності оплати мітка залишається включеною і при спробі залишити магазин зчитувачі на воротах розпізнають радіомітку, після чого відбувається подача сигналу тривоги.

23 січня 1973 року Маріо В. Кардульйо зробив заяву на видачу йому першого патенту в США на активну радіочастотну мітку з можливістю перезапису. У цей же рік, каліфорнійський підприємець, Чарльз Волтон, став власником патенту на пасивну радіомітку, яка застосовувалася для відкриття дверей без використання ключа. Карта, всередині якої був транспондер, здійснювала передачу сигналу до приймача, який розташовувався біля дверей.

Після того, як приймач виявляв певний ідентифікаційний номер, що зберігається в пам'яті радіочастотної мітки, двері відчинялися. Уряд США також працював над системами радіочастотної ідентифікації. У 1970-х Департамент Енергетики доручив Національній Лабораторії в Лос-Аламосі створити систему для відстеження пересування ядерних матеріалів при перевезенні. Команда вчених розробила ідею: мітки розміщувалися у вантажівках, а зчитувальні пристрої – на в'їзді в прохідну об'єкта, що охороняється.

Розташована на воротах антена посилювала сигнал транспондеру, що знаходився у вантажівці. Потім відбувалася передача від транспондера певного ідентифікатора та інших даних, наприклад ідентифікатора водія.

Мітка, що знаходилася у вантажівці, ставала активною під час проходження через ворота, а потім надсилала унікальний ідентифікаційний номер та номер водійських прав. Ця система стала використовуватися на

комерційній основі в середині 80-х, після того, як учені з Лос-Аламосу, які працювали над нею, дали дозвіл на створення компанії, яка почала створювати автоматизовані системи з використанням радіочастотної ідентифікації для оплати проїзду. Даний вид систем став дуже поширеним, активно використовувався на мостах, тунелях та дорогах по всьому світу.

Основними компонентами RFID-системи є пристрій, що зчитує, і транспондер (RFID-мітка), який може зберігати невеликий обсяг інформації (близько 2 кілобайт). Дальність зчитування мітки залежить від робочої частоти. За робочою частотою мітки поділяються на LF (125-134 кГц), HF (13,56 МГц), UHF (860-960 МГц).

Мітки діапазону LF мають низьку ціну, але можуть бути зчитані лише з невеликої відстані. Мітки діапазону HF мають низьку ціну, але на відміну від позначок діапазону LF мають більш високу дальність зчитування.

Мітки діапазону UHF мають найвищу дальність реєстрації. Існує можливість шифрування даних радіочастотної мітки та встановлення пароля для доступу до них. Також істотною перевагою технології радіочастотної ідентифікації є можливість систематизації зібраних даних в інформаційній системі, і, отже, збільшення швидкості пошуку інформації. Електронні мітки бувають активними та пасивними. Для таких пристроїв енергія пристрою для зчитування не впливає на дальність зчитування. Пасивні мітки не мають джерела живлення, тому живляться від енергії електромагнітного сигналу, який поширює зчитувач.

Для даного типу міток дальність зчитування міток залежить від частоти, на якій працює пристрій, що зчитує. Кожен із цих видів пристроїв характеризується своїми перевагами та недоліками.

Пасивні мітки хороші своїм великим терміном експлуатації, а також дешевизною порівняно зі своїм активним аналогом. До того ж, пасивні ідентифікуючі пристрої не потребують заміни елементів живлення. Недоліком пристрою є необхідність використання більш потужних зчитувачів.

На відміну від пасивних міток, активні мають високу дальність зчитування даних, мають можливість бути розпізнаними при русі об'єкта з радіоміткою на великій швидкості щодо пристрою, що зчитує. Недоліком даного типу міток є громіздкість та висока ціна. Таким чином, вибір робочої частоти і типу мітки повинен виконуватися виходячи з поставлених цілей та завдань, які вирішуються в досліджуваній предметній області.

1.4 Огляд області використання RFID технологій

1.4.1 Сфера застосування RFID

Технологія користується попитом у тих галузях, де потрібна здатність функціонування в жорстких умовах, контроль пересування об'єктів у режимі реального часу, відсутність помилок, застосування сучасних рішень для автоматизації, довговічність і швидкість. Щодня з'являється інформація про нові способи застосування цієї технології.

Пристрої обмінюватись даними з радіомітками бувають кількох типів. Зчитувальні пристрої можна розділити на 2 групи: стаціонарні та переносні.

1.4.2 Стаціонарні зчитувачі

Стаціонарні зчитувачі можуть бути закріплені на стінах, порталних воротах і будь-яких інших місцях, де не потрібно переміщення пристрою. Вони виконуються у вигляді воріт, монтуються в стіл або закріплюються поруч із конвеєром на шляху проходження виробів. На відміну від переносних, зчитувачі даного типу мають можливість читати дані у більшому радіусі, мають більшу потужність і можуть обробляти дані з великої кількості міток одночасно.

Такі зчитувачі пов'язані з комп'ютером, де встановлено спеціалізоване

програмне забезпечення контролю та обліку лічених даних. Завдання таких зчитувачів – поетапна фіксація переміщення маркованих об'єктів у реальному часі.

1.4.3 Мобільні зчитувачі

Мобільний вид зчитувальних пристроїв має меншу дальність дії і часто не має постійного зв'язку з програмними засобами контролю та обліку. У внутрішній пам'яті мобільних зчитувачів містяться дані, одержувані з прочитаних міток (надалі отримана інформація може бути передана на комп'ютер)

Даний тип зчитувачів, так само як і стаціонарні, має можливість записувати дані в радіомітку. Відстань, при якому спостерігається гарантований стійкий запис та зчитування даних, залежить від робочої частоти зчитувача та мітки.

1.4.4 Портальні зчитувачі

Портальний пристрій для зчитування використовується для виявлення RFID-міток в проходах, які знаходяться в його радіусі дії. Даний вид тип зчитувача включає наступні компоненти: дві вертикальні стійки, всередині яких розміщені 4 антени, по 2 в кожній стійки. Дані антени приймають та надсилають сигнали.

Кожна стійка покриває антенами певну зону таким чином, щоб перекривалася зона дії другої стійки. Разом вони покривають ділянку між стійками. Стійки виготовляються із матеріалу, який пропускає радіохвилі. Далі будуть розглянуті деякі області і застосування технології RFID.

1.4.5 Використання RFID-систем для автоматизації роботи бібліотек

Експлуатація RFID-систем з метою автоматизації діяльності бібліотек інтенсивно розвивається у багатьох країнах та забезпечує ефективну підтримку процесів, що відбуваються у них.

Раніше системи, що використовуються в цій сфері, зараз вже не відповідають сучасним запитам читачів та бібліотек. Застосування спеціальних зчитувачів та постачання книг бібліотечного фонду RFIDмітками забезпечує контроль над процесами пов'язаними з переміщенням книг приміщеннями бібліотек, скорочує час пошуку необхідного екземпляра серед інших.

Впровадження спеціальних засобів з використанням RFID-технології бібліотеки та електронні квитки читачів дають читачам можливість самостійно проводити здачу та отримання книг, забезпечують більш просту роботу співробітникам бібліотек, допомагають викоринити помилки та неточності, які можуть виникнути при видачі, пошуку та прийомі книг. Також використання радіочастотної технології у воротах на виході із приміщень бібліотеки дозволяє не допустити несанкціонованого викрадення власності бібліотеки та книг, тобто виконує функцію захисту від крадіжок.

Впровадження RFID-системи – затребуваний напрямок автоматизації виробничих процесів на підприємствах. Використання RFIDтехнологій впливає загалом на внутрішньовиробничі процеси, у результаті підвищуються результати роботи всього підприємства.

В автоматизації виробництва технологія RFID використовується у таких напрямках:

- системи контролю виробничого циклу на конвеєрах, RFIDмітка використовується як електронний паспорт об'єкта та операцій;
- логістика таварних запасів, облік використовуваної сировини, марковані RFID-мітками об'єкти враховуються за будь-яких переміщеннях;
- персоніфікація операцій на конвеєрі, до RFID-мітки заносяться дані про те, хто працював з маркованим об'єктом;

- оптимізація витрат від простою, моніторинг фактичних термінів виробництва, система спостереження за виробничою лінією у реальному часі;
- контроль використання підроблених комплектуючих;
- захист репутації у ситуаціях, викликаних використанням неоригінальних комплектуючих;
- оптимізація складського господарства;
- реалізація принципів FIFO.

1.4.6 RFID в логістиці

Для управління ланцюжками поставок потрібна точна інформація. Інтегрування RFID-систем у MES-системи це оптимізації вихідної та вхідної логістики. Складська логістика – галузь, на яку було розроблено перші системи радіочастотної ідентифікації.

З використанням інтегрованої системи товар чітко контролюється всіх рівнях просування: прийом, розміщення, видача. Використання маркування товарів RFID мітками полегшує процес інвентаризації складу.

1.4.7 RFID-система дозволяє

- вести моніторинг ланцюжки поставок у реальному часі (за допомогою web-доступу);
- відстежувати товар на кожному етапі ланцюжка поставок;

1.4.8 RFID для інвентаризації основних сировищ

Основні засоби підприємства маркуються RFID-мітками. Сам процес інвентаризації здійснюється за допомогою ручного зчитувача RFID. Він здатний зчитувати інформацію про предмети, промарковані RFID-мітками. Зчитування займає секунди (до 100 одиниць за секунду). Інформація про об'єкти передається до системи та автоматично порівнює отримані дані з інформацією. Опрацювавши результати, система автоматично формує звіти. Форма звіту може бути різною: складання загальних списків основних засобів, що знаходяться у приміщеннях, списки об'єктів, які не були виявлені.

1.4.9 Можливості RFID-системи:

- облік руху основних засобів та товарно-матеріальних цінностей;
- контроль переміщення основних засобів та товарно-матеріальних цінностей;

1.5 Порівняльний аналіз радіочастотної ідентифікації з іншими технологіями

Іншою поширеною технологією автоматичної ідентифікації є використання штрих-кодів. Ця технологія заснована на використанні хвиль світла. Штрих-коди, також відомі як оптичні мітки, застосовуються протягом багатьох років для маркування товарів. Але використання даної технології ідентифікації має цілу низку недоліків, які звужують область її застосування.:

- невелика ємкість (одномірний штрих-код розміром з аркуш паперу формату «А4» дає змогу закодувати приблизно 50 байт інформації);
- відсутність способу змінити записані дані (штрих-коди відносяться до типу міток, які можуть бути використані тільки для читання даних);
- повільна швидкість збору інформації;

- маленька надійність;
- недовговічність – термін експлуатації залежить від різних характеристик, таких як об'єкт, на якому був розташований штрих-код, від умов навколишнього середовища, від фарби, що використовується під час створення коду.

Вище перелічені недоліки відсутні системи автоматичної ідентифікації з використанням технології RFID. Порівняння штрих-кодів та радіоміток представлено в таблиці 1.1

Таблиця 1.1 – Зрівняння характеристик штрих-кодів та радіоміток

Параметр	Штрих-коди	Радіомітки
Максимальний об'єм інформації, Кбайт	2	32
Можливість зміну інформації	–	+
Можливість автоматичного зчитування інформації	+	+
Максимальна дистанція до пристрою збору даних	0,5	5*
Вплив точність позиції етикетки на момент отримання сканеру	Високе	Низьке
Швидкість зчитування інформації	Середня	Дуже висока
Вплив на читабельність коду	Критичні оптичні характеристики	Критична електромагнітна проникність матеріалу
Возможність візуальної оцінки кода	Обмежена	–
Складність підробки	Низька	Висока
Вартість	Низька	Висока

На основі проведеного порівняльного аналізу можна дійти до невтішного висновку, що з досліджуваної предметної області технологія радіочастотної ідентифікації є найбільш підходящою.

— Важливим аспектом використання технології радіочастотної ідентифікації є обсяг даних, який зможе зберігати радіомітка. Розглянуто модель приладу обліку електроенергії – лічильник «Меркурій 236 ART». У ході проведеного аналізу виявлено найбільш значущі характеристики, які необхідно зберігати на радіочастотній мітці: унікальний ідентифікаційний номер;

- назва;
- виробник;
- дата вводу в експлуатацію;
- дата останньої перевірки (облік даних);
- Унікальний ідентифікаційний номер перевіряючого працівника; -

Свідчення приладу (кВт·ч).

Для зберігання вищезазначених параметрів пам'ять радіочастотної мітки повинна бути від 512 байт-2 кбайт.

Дана модель лічильника також має додаткові характеристики, які є важливими та можуть бути записані на радіочастотну мітку:

- фазна напруга та струм у кожній фазі;
- коефіцієнт потужності по кожній фазі і сумарний по трьом фаз;
- поточний час та дату;
- температуру у корпусі;
- дату та час роботи електронних пломб;
- дату и время доступа по цифровым интерфейсам;

1.6 Підсумки до розділу

У першому розділі було проведено аналіз предметної галузі, існуючих завдань у сфері збору даних. Було здійснено огляд технології радіочастотної ідентифікації та областей її застосування. Визначено переваги цієї технології над іншими технологіями автоматичної ідентифікації. Обґрунтовано можливість використання даної технології в досліджуваній предметній галузі.

В результаті проведеного аналізу можна зробити висновок про те, що існуючі програмні засоби не вирішують ряд важливих завдань: просте розгортання системи та її сервісне обслуговування, початкове формування відомостей про прилади обліку після їх встановлення, зміна даних та переконфігурування системи при додаванні/заміні/видаленні приладів обліку.

Програмні засоби, що розробляються, допоможуть вирішити зазначені вище проблеми, використання технології радіочастотної ідентифікації дозволить скоротити витрати на створення та інжинерінг систем збору інформації, скоротити час, що витрачається на збір даних та конфігурування системи, буде спрощено розгортання системи та її сервісне обслуговування

2. РОЗРОБКА ОНТОЛОГІЧНОЇ МОДЕЛІ СИСТЕМИ ЗБОРУ ДАНИХ ЕНЕРГОСПОЖИВАННЯ ЗА ДОПОМОГОЮ ТЕХНОЛОГІЇ RFID

2.1 Використання онтології для дослідження предметної області

Для систематизації та формалізації предметної області запропоновано використання онтології.

На формальному рівні онтологія - це система, яка складається з певного набору понять і різних тверджень про ці поняття, на основі яких можна створювати відносини, класи, об'єкти, теорії і функції.

Онтологія включає:

1. система понять предметної області, в якій формулюється і інтерпретується емпірична теорія;
2. властивості, атрибути, величини і відповідні вимірювальні процедури, що інтерпретуються в системі понять;
3. апріорні та експертні знання;
4. знання, інтерпретовані в системі понять програмного забезпечення, отримані в процесі побудови логічних, кількісних і конструктивних емпіричних теорій.
5. Процес онтологічного (концептуального) моделювання передбачає:
6. вибір експертом понять, значущих для досліджуваної предметної області;
7. категоризація (початкова класифікація) вибраних понять;
8. подальша систематизація за категоріями – таксономічна (ієрархічна) організація понятійної лексики;

Переваги використання онтології в розробці програмного забезпечення:

1. формування та закріплення спільних для всіх фахівців знань про предметну область;
2. Явна концептуалізація предметної області, що дозволяє описати семантику даних.
3. Інтеграція та спільне використання різномірних даних та знань в межах однієї системи;

2.2 Побудова метаонтології програмних засобів

Розробка систем збору даних про енергоспоживання вимагає детального вивчення і аналізу предметної області. Необхідно підготувати чітку модель, на основі якої буде розроблятися система. Запропоновано використовувати онтологічний аналіз при розробці програмних засобів збору даних про енергоспоживання.

Метаонтологія - це онтологія, яка оперує загальними поняттями і відносинами. Поняттями на метарівні є «об'єкт», «властивість», тобто .

Перед розробкою програмних засобів збору даних про енергоспоживання необхідно провести детальний аналіз предметної області, виявити суб'єкти предметної області, визначити функції, які буде виконувати розроблена система.

Для систематизації та формалізації предметної області збору даних про енергоспоживання запропоновано використання онтології.

Побудова онтології дозволяє вирішити наступні завдання:

1. – виділити значущі характеристики і поняття досліджуваної предметної області;
2. класифікувати та систематизувати вибрані поняття;
3. систематизувати характеристики предметної області у вигляді ієрархії; – визначення відношень між поняттями («частина-ціле»,

"є підкласом", "має ефект", "схоже" і так далі).

Для розроблюваної системи були виділені наступні поняття (концепції):

4. RFID-зчитувач - це пристрій, який буде використовуватися для

Пишіть і читайте етикетки.

5. мітка – носій інформації, який взаємодіє зі зчитувачем на основі технології радіочастотної ідентифікації;

6. Дані – інформація, яка буде описувати характеристики приладу (унікальний ідентифікатор, найменування, виробник, дата введення в експлуатацію, дата останнього зчитування, унікальний ідентифікатор співробітника, який проводить перевірку, показання приладу);

7. пристрій – гаджет, до якого підключається RFID-зчитувач (комп'ютер, планшет, смартфон);

8. Контролер – особа, яка перевіряє прилад, знімає показання з бирки або вносить дані на бирку:

9. application – програмне забезпечення, що дозволяє зчитувачу взаємодіяти з міткою, записувати та зчитувати дані;

10. Сховище даних – це місце на носії даних, яке може бути представлено у вигляді структурованого файлу (наприклад, XML-файлу) або бази даних.

11. Показання приладу - це деяка величина, яка показує показання споживання електроенергії.

Для побудови онтології використовувався редактор Protege Вибрані класи додано на вкладку Entities->ClassHierarchy.

Створені класи показані на рисунку 2.1.

Для опису зв'язків між обраними класами було створено 9 властивостей. Кожна властивість описує певні дії, що відбуваються між класами онтологій. Створені властивості наведені в таблиці 2.1.

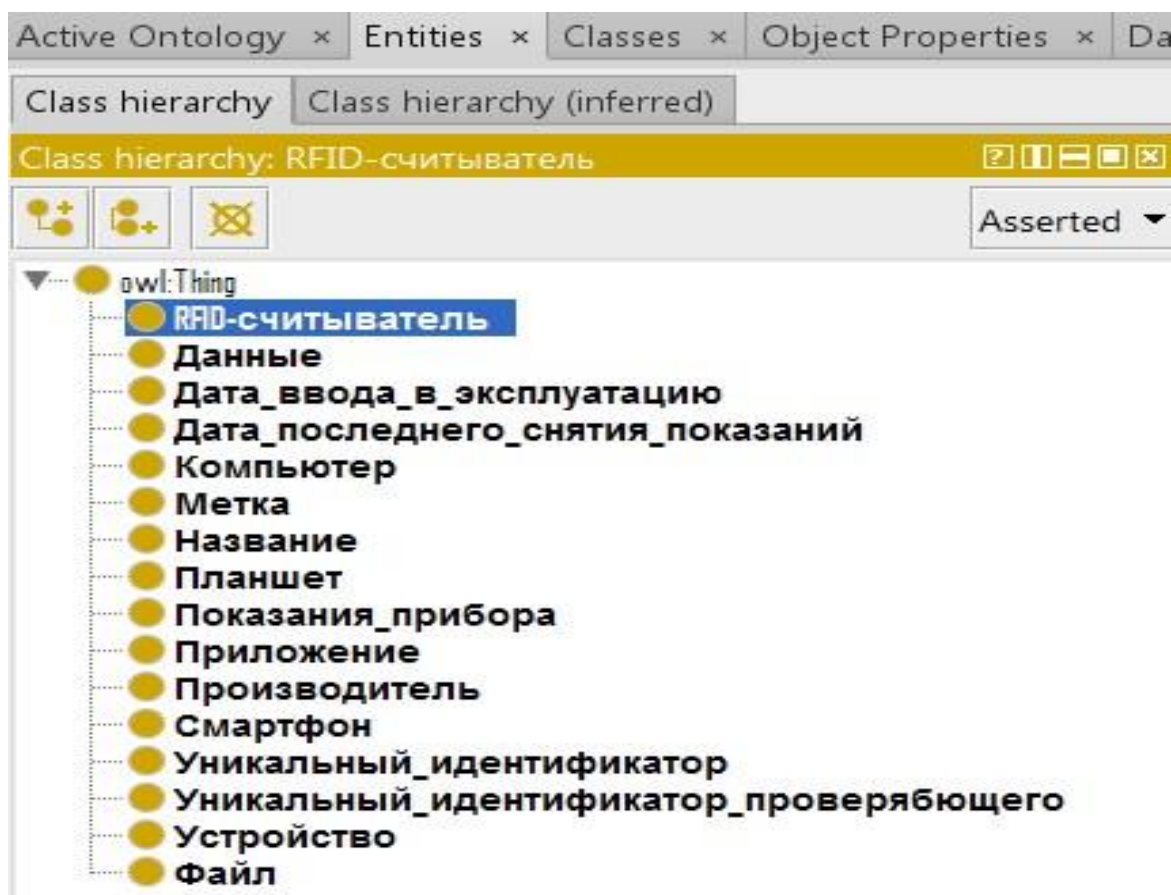


Рисунок 2.1 – Класика, створена в редакторі онтології Protégé

Таблица 2.1 – Властивості, описані в редакторі протеге

№	Ім'я об'єкта	Значення властивості (відношення)
1	connectedTo	"Connected to" – показує зв'язок між об'єктами
2	є	"Is" – вказує на те, що певний клас є підкласом іншого класу
3	isTypeOf	"Є типом" – вказує на те, що певний клас є підтипом Інший клас
4	readsDataFrom	"Read Data From" — вказує на те, що об'єкт одного класу зчитує дані з об'єкта іншого класу
5	returnsДані	«Повертає дані» — вказує на те, що об'єкт одного класу повертає (витягує) дані з об'єкта іншого класу
6	runsOn	Run On: вказує на те, що об'єкт одного класу є програмним забезпеченням і виконується на об'єкті іншого класу
7	savesDataTo	Зберігає дані в: вказує на те, що об'єкт одного класу зберігає дані для об'єкта іншого класу

Продовження таблиці 2.1

№	Ім'я об'єкта	Значення властивості (відношення)
8	sendsЗапит	"Надсилає запит" — вказує на те, що об'єкт одного класу надсилає запит об'єкту іншого класу
9	Магазинах	"Stores" - вказує на те, що об'єкт одного класу зберігає об'єкт іншого класу

Користувач також може вибрати кілька класів для зв'язку за допомогою властивостей. Властивості представлені в полі "Властивості об'єкта-
>owl:topObjectProperty».

Зв'язки між класами вказуються в блоці "Домени та діапазони". Щоб задати напрямок зв'язків між класами, потрібно в блоці "Домен" вказати клас, з якого походить зв'язок. У блоці «Діапазони (перетин)» потрібно вказати клас, на який спрямоване посилення.

На рисунку 2.2 показаний приклад того, як описувати відношення класів в Protege.

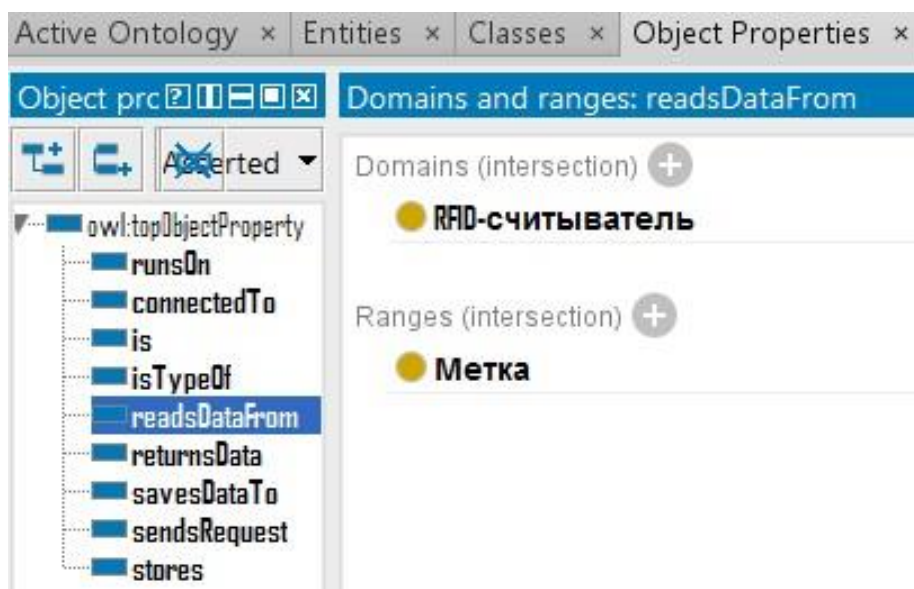


Рисунок 2.2 – Опис класових зв'язків у Protege

Розроблено та візуалізовано онтологічну модель для програмного забезпечення для збору даних про енергоспоживання. Для візуалізації

використовувався плагін OntoGraf. Візуалізація проекрованої моделі наведена на рисунку 2.3.

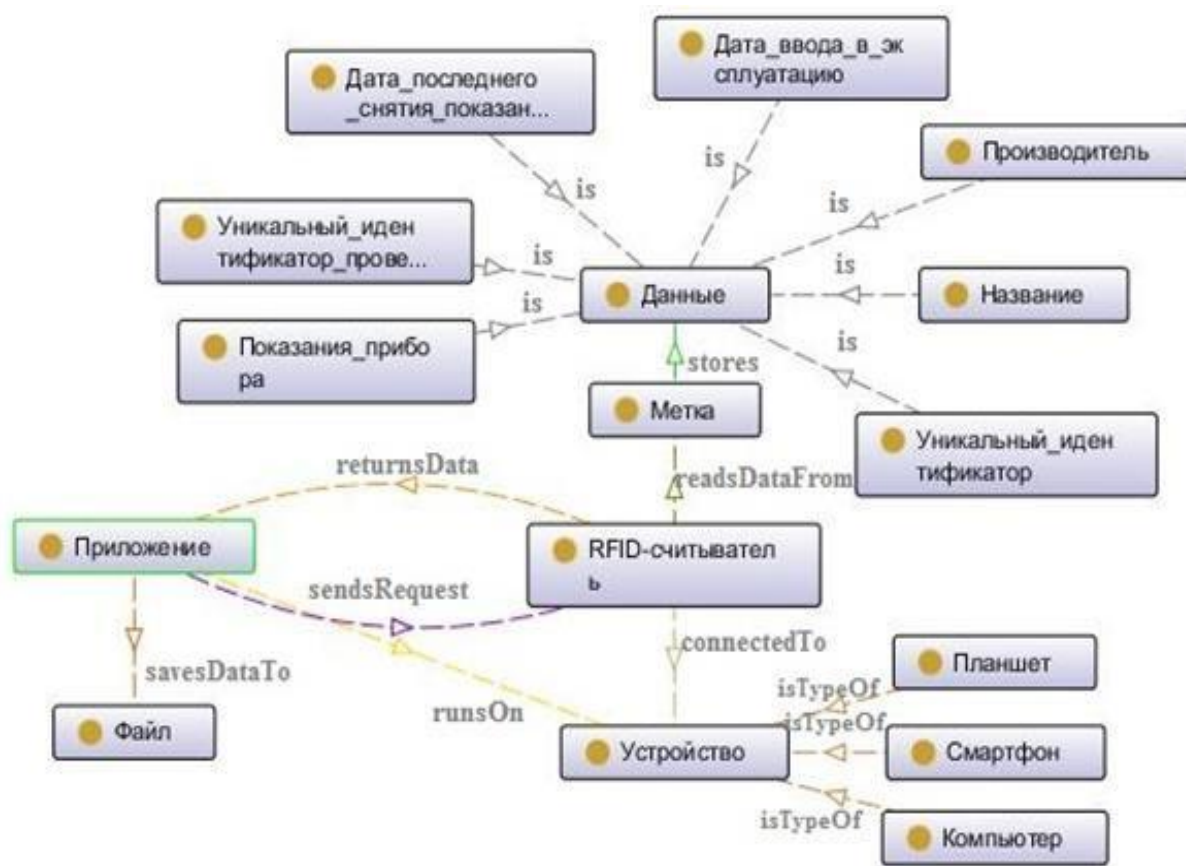


Рисунок 2.3 – Метаонтологія програмних засобів

Побудована метаонтологія являє собою об'єкти досліджуваної предметної області, а також відносини між ними. Зв'язки між класами позначаються стрілками. Мітка над стрілкою відповідає назві зв'язку, який пов'язує класи.

2.3 Побудова онтології програмних засобів

Створена метаонтологія містить об'єкти реального світу і докладно описує досліджувану предметну область. Для розробки програмних засобів необхідно абстрагуватися і виключити зайві класи з онтологічної моделі, перейти до спрощеної моделі, на основі якої будуть проектуватися програмні засоби.

Створену модель можна спростити наступним чином: раніше визначені класи даних про лічильник можуть бути перетворені у властивості класу "Item".

Також пропонується ввести клас, що відповідає за зберігання даних про об'єкти класу "Device" - "ListOfItems". Зчитувач був описаний за допомогою «RFID-зчитувача».

У програмній моделі не буде класів об'єктів реального світу, таких як:

1. марка;
2. дані; – додаток; – пристрій.

Додано новий клас "Verifactor", який відповідає за перевірку

Значення параметрів приладу обліку:

3. дата останньої перевірки; – дата введення в експлуатацію;
4. значення показань приладу.

Спрощена онтологія програмних засобів представлена на рисунку 2.4

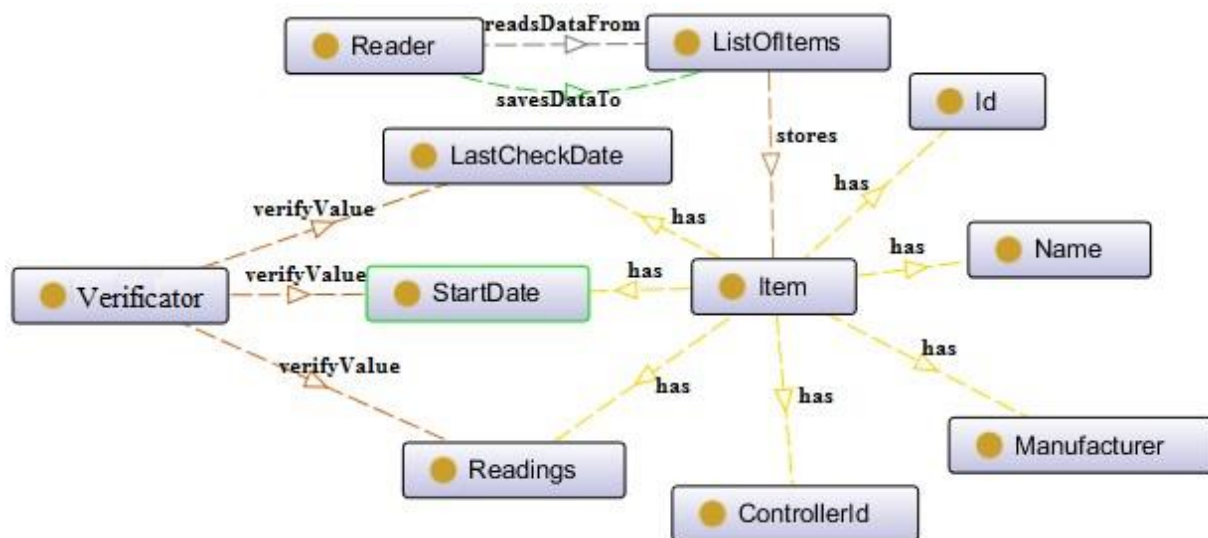


Рисунок 2.4 – Онтологія програмних засобів

Таким чином, створена онтологічна модель може бути використана для подальшої розробки програмних засобів збору даних про споживання енергії з використанням технології радіочастотної ідентифікації.

На основі побудованої онтологічної моделі може бути сформований код у форматі RDF/XML. Нижче наведено фрагмент коду, який описує класи онтологій та взаємозв'язки між ними.

```

<owl:ObjectProperty
  rdf:about="http://www.semanticweb.org/user/ontologies/2017/3/RF
ID_
  ONTOLOGY#containsFields">          <rdfs:subPropertyOf
  rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/
>
      <rdfs:domain
  rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3
/RF
ID_ONTOLOGY#Item"/>          <rdfs:range
  rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3
/RF
ID_ONTOLOGY#ControllerId"/>          <rdfs:range
  rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3
/RF
ID_ONTOLOGY#Id"/>          <rdfs:range
  rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3
/RF
ID_ONTOLOGY#LastCheckDate"/>          <rdfs:range
  rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3
/RF ID_ONTOLOGY#Manufacturer"/>          <rdfs:range
  rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3
/RF
ID_ONTOLOGY#Name"/>          <rdfs:range
  rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3
/RF
ID_ONTOLOGY#Readings"/>          <rdfs:range

```

```
    rdf:resource="http://www.semanticweb.org/user/ontologies/2017/3  
/RF  
ID_ONTOLOGY#StartDate"/>  
    </owl:ObjectProperty>
```

3 РОЗРОБКА МЕТОДІВ ВЕРИФІКАЦІЇ ТА ВАЛІДАЦІЇ ДЛЯ АНАЛІЗУ

3.1 Дані обліку енергоресурсів

Важливим аспектом є перевірка даних, отриманих безпосередньо від приладів обліку енергії. Верифікація - це перевірка отриманого значення технологічної мітки на достовірність. Валідація - це заміна отриманого недостовірною значення бирки на достовірне значення.

Наприклад, технічний персонал неправильно підключив лічильник. В результаті значення оперативної мітки, що відповідає за виміряне значення параметра енергоресурсу, буде невизначеним (в автоматизованій системі збору даних воно дорівнює NaN – Not a Number).

Використання такої величини в процесах обробки даних, таких як білінгові системи, призведе до помилок і збоїв при формуванні рахунків, звітів і розрахунку агрегованих показників енергоспоживання. Нижче розглянемо можливі алгоритми перевірки.

Алгоритм перевірки пов'язаного параметра.

Цей алгоритм дозволяє порівнювати значення вимірюваного параметра енергоспоживання зі стороннім параметром. Нижче наведена формула перевірки відповідного параметра.

P_c - поточне значення пов'язаного з ним параметра;

P_p - попереднє значення пов'язаного параметра;

St - знак зв'язку (може мати значення 0 або 1).

Якщо значення St дорівнює нулю, застосовується прямо пропорційне правило зчеплення. Якщо величина St дорівнює одиниці, застосовується обернена пропорційна зв'язок.

Розглянемо застосування алгоритму на прикладі. Для визначення достовірності значення параметра процесу використовуються пов'язані значення,

що зберігаються в базі даних (значення в базі даних можуть бути сформовані сторонньою або резервною системою).

"Tag1" - значення основного параметра обробки даних, "Tag2" - значення, визначене в базі даних. Для кожного отриманого значення "Tag1" алгоритм звертається до необхідної бази даних та зчитує значення "Tag2".

Потім, з урахуванням типу підключення, поточні значення цих параметрів порівнюються з їх попередніми значеннями, і в результаті встановлюється надійність значення «Tag1».

Алгоритм перевірки за інтегральною величиною.

Цей алгоритм призначений для визначення достовірності поточного значення параметра інтегрального типу (наприклад, обсягу споживаної води, потужності спожитої електричної енергії). Нижче наводиться формула для перевірки за інтегральною величиною (3.1):

$$V_v = T_c \geq T_p. \quad (3.1)$$

Поточне значення вважається дійсним, якщо воно дорівнює або більше попереднього значення, в іншому випадку значення вважається недостовірним. Цей алгоритм не має конфігураційних параметрів, оскільки при довірчому розрахунку використовуються тільки поточні та попередні значення параметра.

Алгоритм валідації за останнім достовірним значенням.

Цей алгоритм замінює ненадійне значення параметра останнім достовірним значенням, яке зберігається в контексті аналізу. Можливо, що серед отриманих значень параметра немає достовірних значень.

У цьому випадку цей алгоритм не замінить ненадійні значення і згенерує попереджувальне повідомлення в журналі подій.

Така ситуація найбільш ймовірна при першому запуску автоматизованої системи, коли відсутні реальні достовірні дані. Цей алгоритм не має параметрів конфігурації.

У таблиці 3.1 представлені методи перевірки, адаптовані до досліджуваної предметної області.

Таблиця 3.1 – Методи перевірки

Подія	Перевірене значення	Умова	Результат
Запис/читання даних з тегу	Показання приладу	Поточне значення зчитування $> =$ попереднє значення	Верифікація успішна
	Дата введення в експлуатацію	дата введення в експлуатацію $\leq$ дата останнього читання	Верифікація успішна
Запис/читання даних з тегу	Дата останнього читання	Дата останнього читання $\geq$ Дата введення в експлуатацію	Верифікація успішна
Запис даних	Всі параметри	Відсутній хоча б один параметр	Заборонити економію

Також, якщо немає можливості зібрати дані, рекомендується усереднити отримані раніше значення (квартальні, місячні, щотижневі і т.д.) до необхідного рівня вибірки і записати це значення в базу даних .

Нижче наведена формула (3) для обчислення середнього значення:

$$x_{\text{ср.}} = \frac{1}{T} * x \quad (3.2)$$

x – значення коефіцієнта інформаційної бази даних з періодом дискретизації T ;

T - період дискретизації фактора x ; $x_{\text{ср.}}$ – добове усереднення фактора X .

Такий підхід дозволить врахувати всі витрати на енергоресурси, але призведе до зниження якості математичної моделі в цілому, придатність якої буде визначена після перевірки отриманих результатів.

Максимальна похибка технічних розрахунків з метою моделювання режимів енергоспоживання не повинна перевищувати п'яти відсотків, тоді необхідний обсяг вибірки буде описуватися щоденною статистикою, що охоплює 1 рік (365 днів).

3.2 Висновки до розділу

У третьому розділі предметна область була досліджена за допомогою методу онтологічного аналізу.

Побудовано метаонтологію предметної області. Потім була побудована спрощена онтологічна модель.

На основі створеної моделі може бути побудована архітектура розроблених програмних засобів.

Розроблені методи верифікації та валідації для перевірки значень, які зчитуються з етикетки та записуються на неї.

Аналіз вимог

Системні вимоги:

1. Пристрій має USB-порт;
2. операційна система Windows 7 і вище;
3. LotusCardDriver.dll — драйвер для взаємодії з зчитувачем, .NET Framework версії 4.0 і новіших.

Розроблені варіанти використання програмного забезпечення представлені в таблиці 3.2.

Таблиця 3.2 – Опис варіантів використання

Ім'я	Дія
Зчитування даних	Дані зчитуються з радіомітки і відображаються в графічному вигляді Інтерфейс користувача
Перевірити	Перевіряється прочитані/підготовлені до запису дані. Аналізується правильність показань, введення в експлуатацію і дати початку Використання
Порівняння даних із попередніми даними	Дані порівнюються з попередніми Записані
Запис даних	Дані записуються на радіомітку, статус запису відображається в графічному користувачеві Інтерфейс

Є 3 «актори»: користувач, RFID-мітка, автоматизована система обліку енергії. Схема варіантів використання програмного забезпечення показана на рисунку 3.1.

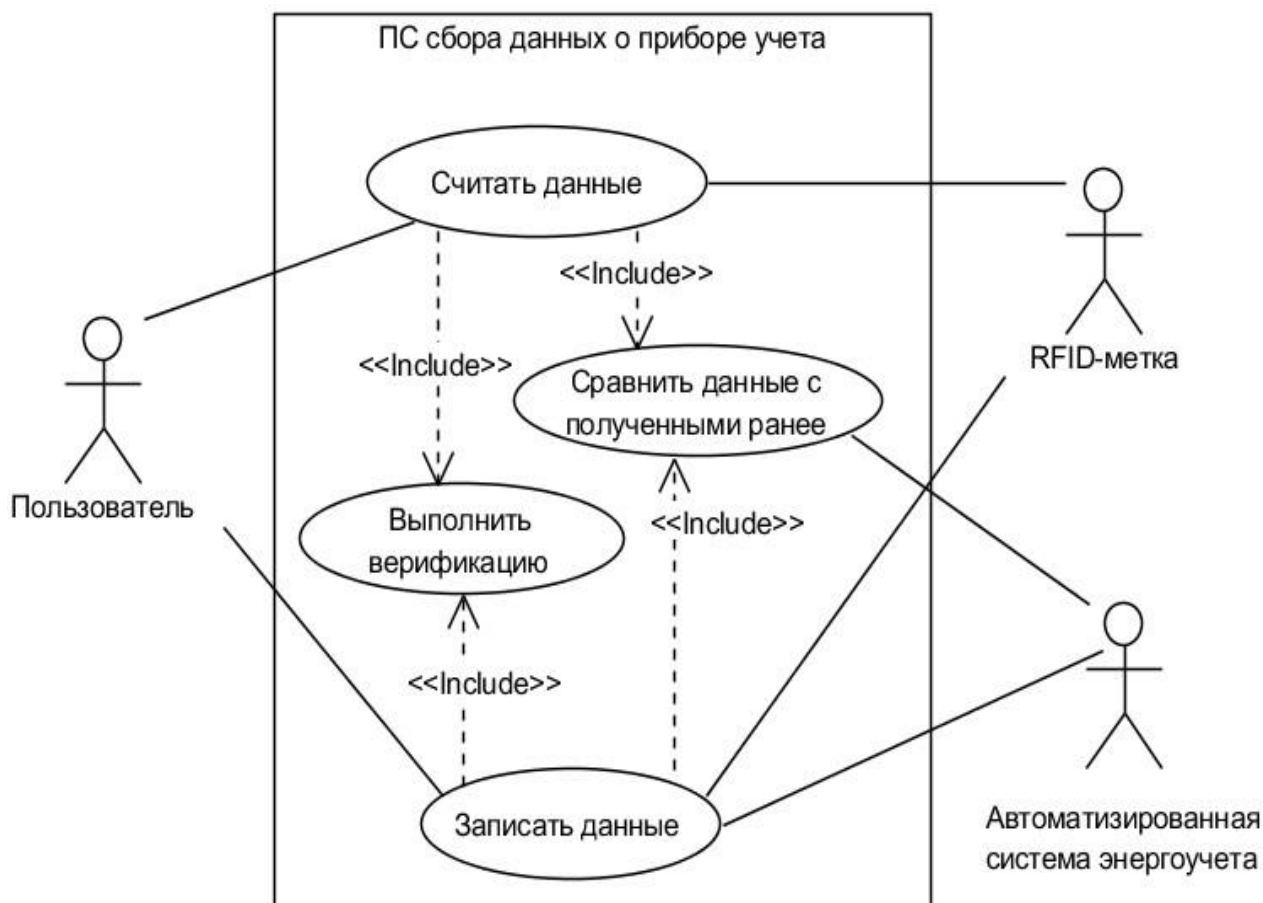


Рисунок 3.1 – Діаграма варіантів використання програмного забезпечення

Прочитані дані та порівняння даних із раніше зібраними випадками використання включено до параметра «Читання даних». Варіант використання «Порівняти дані з раніше отриманими» включено в опцію «Записати дані».

З точки зору окремого прецеденту виділяють два типи суб'єктів:

1. головні дійові особи – суб'єкти, які ініціюють прецедент;
2. Вторинні суб'єкти - це суб'єкти, які взаємодіють з прецедентом після його ініціації .

Сценарій використання читаних даних обговорюється в таблиці 3.3

Таблиця 3.3 – Сценарій використання читаних даних

Ідентифікатор: 1
Резюме: Користувач зчитує дані з етикетки
Головні дійові особи: Користувач
Другорядні дійові особи: RFID-мітка
Попередні вимоги: немає
Основний потік: 1. Користувач натискає на кнопку «Прочитати дані» 2. Якщо мітка містить дійсні дані 1. Дані відображаються в полях головного вікна програми 2. Журнал містить інформацію про успішне зчитування даних 3. Якщо етикетка не містить даних 1. З'явиться повідомлення про помилку читання 2. Поля залишаються порожніми, дані не відображаються
Умови запису: немає
Альтернативні потоки: Немає

Сценарій використання «Повної перевірки» розглянуто в таблиці 3.4

Таблиця 3.4 – Сценарій для варіанту використання «Виповнити верифікацію»

Приклад використання: "Повна перевірка"
ID: 2
Підсумок: користувач зчитує дані з мітки або намагається записати дані в мітку..
Головні дійові особи: Користувач
Другорядні дійові особи: RFID-мітка
Обов'язкові умови: користувач натискає на кнопку «Прочитати дані» або на кнопку «Записати дані»
Основні: 1. Якщо на етикетці вказані дійсні дати та показання 1. Верифікація пройшла успішно 2. Дані виводяться в поля/записуються в мітку 2. Якщо на етикетці вказані неправильні дати або показання 1. З'явиться повідомлення про те, що значення дати неправильні 1.1 Виводиться повідомлення про неправильне значення показань
Умови запису: немає
Альтернативні потоки: Немає

Сценарій використання «Виконати перевірку» обговорюється в таблиці 3.5.

Таблиця 3.5 – Сценарій для варіанта використання верифікації

Приклад використання: "Порівняйте дані з попередніми"
Ідентифікатор: 3
Підсумок: користувач зчитує дані з мітки або намагається записати дані в мітку.
Головні дійові особи: Користувач
Другорядні дійові особи: 1. RFID-мітка; – Автоматизована система обліку енергії.
Обов'язкові умови: користувач натискає на кнопку «Прочитати дані» або на кнопку «Записати дані»
Основний потік: 1. Якщо зчитані дані не зберігаються в об'єкті 1. Дані про пристрій вважаються новими і будуть записані в сховище 2. Якщо сховище містить інформацію про цей пристрій 1. При спробі записати дані, які точно збігаються з раніше написаними, ви отримаєте повідомлення про те, що такі дані вже містяться в сховищі 2. Дані не записуються в сховище 3. Якщо сховище містить інформацію про цей пристрій, але не повністю збігається з інформацією, що зчитується з тега 1. Триває верифікація даних 1.1 У сховище записуються дані, які відрізняються від даних, які були записані раніше
Умови запису: немає
Альтернативні потоки: Немає

Сценарії використання детально описують послідовність дій, які відбуваються під час взаємодії акторів і сценаріїв використання.

Виходячи з описаних сценаріїв, можуть бути реалізовані наступні:

Актуальні функції програмного забезпечення

На рисунку 3.2 показана діаграма послідовності для опції «Запис даних».

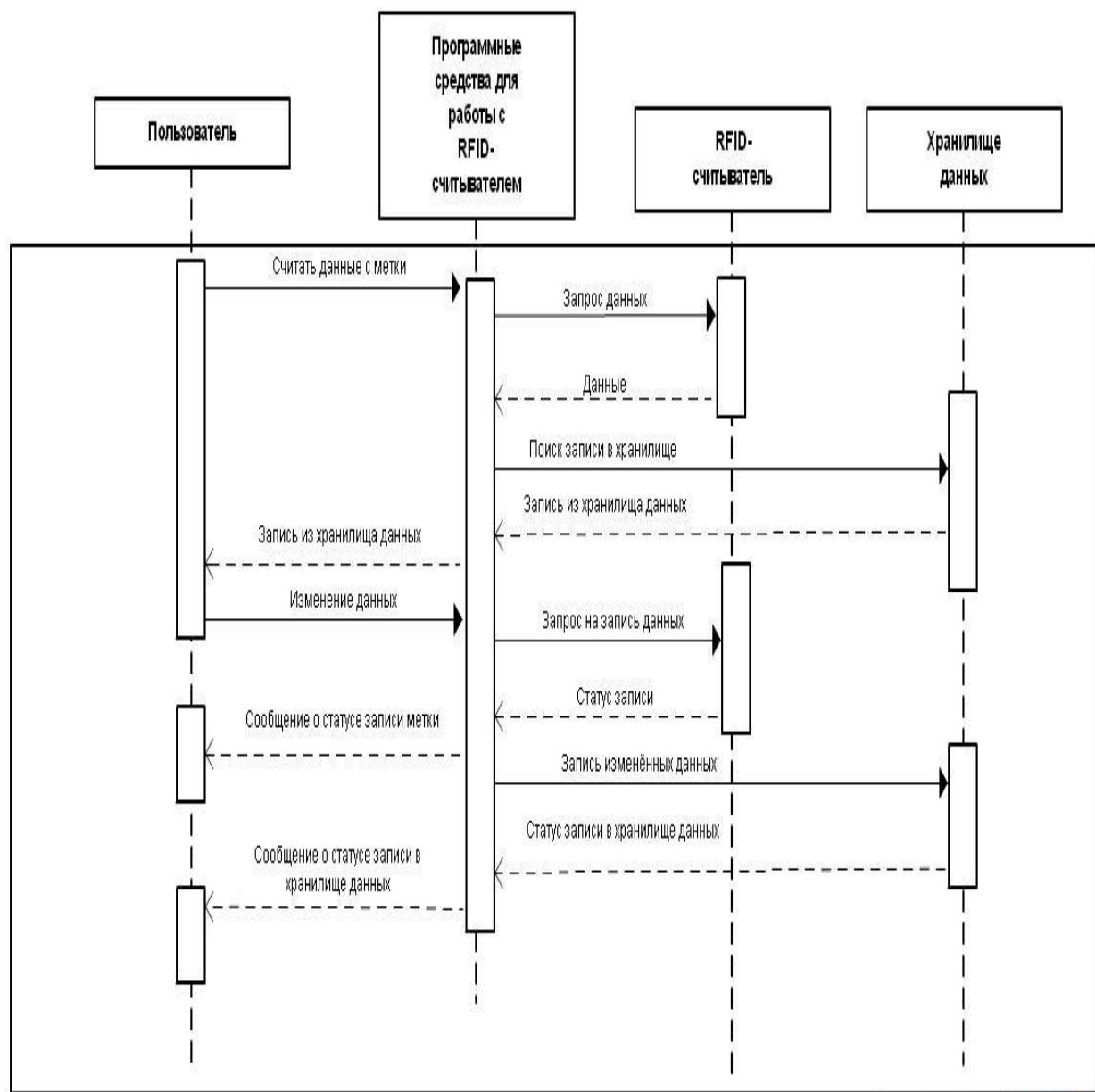


Рисунок 3.2 - Диаграмма последовательности для опции «Запис даних»

Рядок активації показує часовий проміжок активності конкретного суб'єкта. Стрілка з пунктирною лінією вказує на відгуки, які вказують на повернення повідомлень.

3.2 Розробка програмного забезпечення

Для розробки програмних засобів використовувалася мова C#, а для створення сучасного та зручного інтерфейсу користувача – технологія WPF (Windows Presentation Foundation).

В якості сховища даних був обраний xml-файл. Об'єкти класу Item розміщуються в класі list ListOfItems, тоді цей клас може бути серіалізований у вигляді xml файлу.

Структура створеного файлу наведена на рисунку 3.3

```
<?xml version="1.0"?>
<ListOfItems xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd:
  <Item>
    <Id>4105086615</Id>
    <Name>General Electric</Name>
    <Manufacturer>PR-1</Manufacturer>
    <StartDate>1987-09-17T00:00:00</StartDate>
    <LastCheckDate>2000-09-17T00:00:00</LastCheckDate>
    <ControllerId>John71</ControllerId>
    <Readings>0</Readings>
  </Item>
  <Item>
    <Id>1888429585</Id>
    <Name>Ee</Name>
    <Manufacturer>Bb</Manufacturer>
    <StartDate>2012-06-15T00:00:00</StartDate>
    <LastCheckDate>2017-03-28T00:00:00</LastCheckDate>
    <ControllerId>1R</ControllerId>
    <Readings>536</Readings>
  </Item>
</ListOfItems>
```

Рисунок 3.3 – Структура xml-файлу з даними про прилади обліку

Розроблено зручний інтерфейс. Основними елементами є поля для введення/виведення даних: поле ID доступне лише для читання, з використанням унікального ідентифікаційного номера мітки, встановленого виробником.

Поля Name, Manufacturer, StartDate, LastCheckDate, ControllerId, Readings доступні як для читання, так і для запису. Праворуч від полів є кнопка «Очистити всі поля». Під полями даних розташована кнопка «Прочитати дані». У нижній частині вікна є поле виведення «Журнал подій» для відображення інформації про дії, що виконуються під час взаємодії з читачем. Праворуч від полів даних є блок «Статус», який відображає статус операцій. У правому верхньому куті знаходиться блок «Налаштування читання», який містить 2 кнопки.

Перша кнопка дозволяє увімкнути/вимкнути звук читання. Друга кнопка встановлює режим читання. Якщо обраний нескінченний режим читання, то читач буде постійно чекати наближення мітки у вічному циклі, в цьому режимі не потрібно використовувати кнопку для зчитування інформації, можна зрушити кілька міток в сторону, одну за одною. Поруч із кнопкою згортання головного вікна є кнопка довідки, при натисканні на яку виводиться вікно з інструкцією користувача, що пояснює деталі взаємодії з додатком. Інтерфейс користувача розроблених програмних засобів наведено на рисунку 3.4.

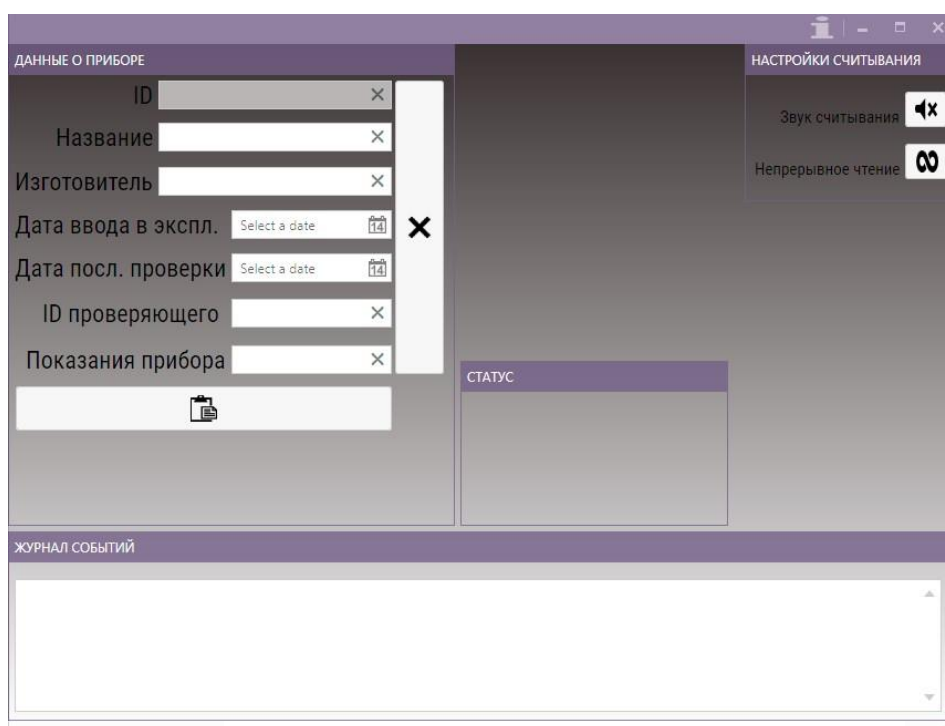


Рисунок 3.4 - Інтерфейс користувача

Перша кнопка дозволяє увімкнути/вимкнути звук читання. Друга кнопка встановлює режим читання. Якщо обраний нескінченний режим читання, то читач буде постійно чекати наближення мітки у вічному циклі, в цьому режимі не потрібно використовувати кнопку для зчитування інформації, можна зрушити кілька міток в сторону, одну за одною. Поруч із кнопкою згортання головного вікна є кнопка довідки, при натисканні на яку виводиться вікно з інструкцією користувача, що пояснює деталі взаємодії з додатком.

Компоненти програмних засобів показані на рисунку 3.5.

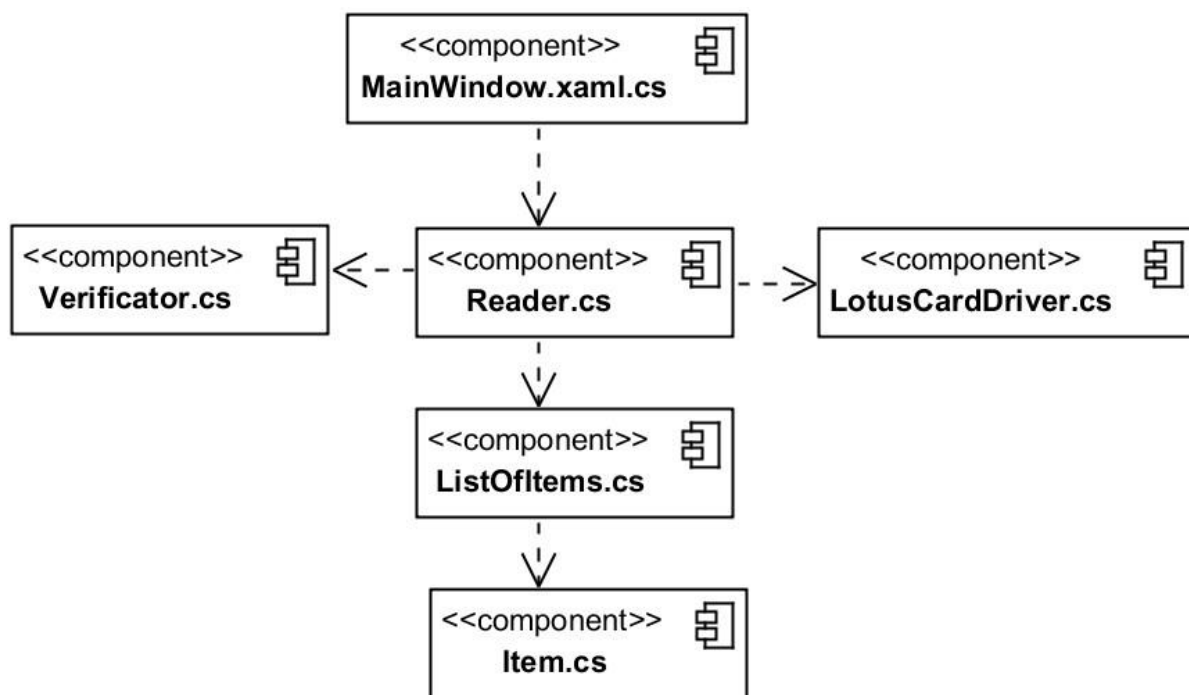


Рисунок 3.5 – Схема компонентів

Детальний опис компонентів наведено в таблиці 3.6

Таблиця 3.6– Опис компонентів

Компонент	Опис
MainWindow.xaml.cs	містить логіку роботи з елементами упуправління головного вікна та методи оброзробки їх подій
Reader.cs	містить логіку взаємодії з читачем
Verificator.cs	містить логіку методів перевірки значень, отриманих з мітки і збережених в її пам'яті
ListOfItems.cs	зберігає об'єкти класу Item у файл та завантажується з файлу
Item.cs	описує дані про лічильник
LotusCardDriver.cs	бібліотека, що надає інтерфейс для роботи з читачем

Далі розглянемо методи деяких основних класів. У таблиці 3.7 наведено детальний опис методів класу.

Таблиця 3.7 – Опис методів

Клас	Метод	Опис
Читач	ReadData (ЧитанняДані)	Зчитує дані з рарадіомітки
	WriteData	Запис даних на рарадіомітку
	Звуковий сигнал	Формує звзвуківий сигнал прпро роботу зчитувача
	AddLog (Журнал додавання)	Додає дані до журналу
Перевірки	VerifyReadingsByBoundedParameter	Перевіряє значення показань за допомогою Пов'язаний параметр
	VerifyStartDate	Перевіряє правильність значення дати, внесеної в Операція
	VerifyLastCheckDate	Перевіряє правильність значення останньої дати Перевіряти
ListOfItems	Навантаження	Завантажує список об'єктів класу Item з XML-файл
	Рятувати	Зберігає список об'єктів класу Item у xml-файл
	GetItemById	Повертає об'єкт класу Item з Специфічні ідентифікатори

На рисунку 3.6 представлена діаграма класів реалізації, згенерована Visual Studio

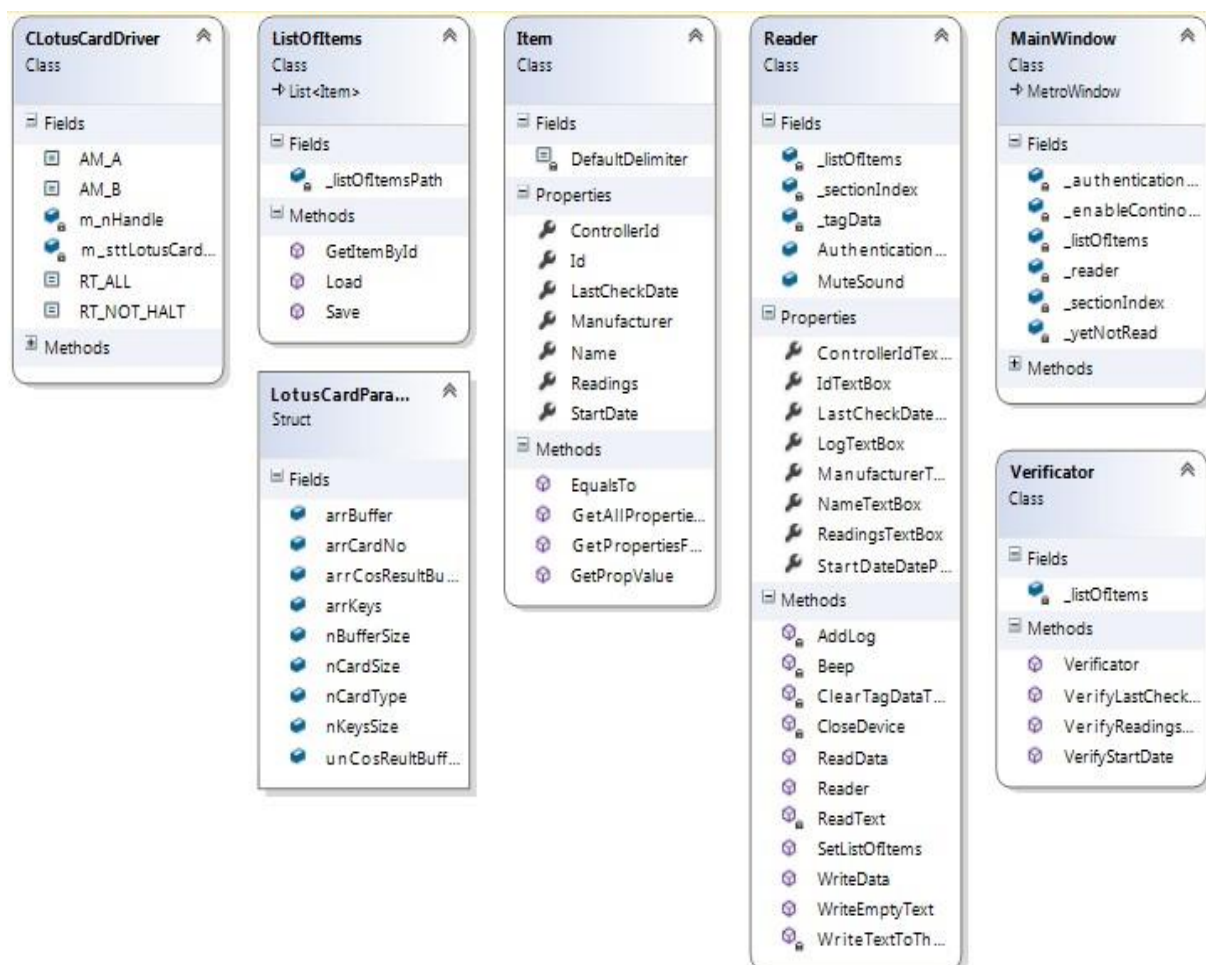


Рисунок 3.6 – Діаграма класів, згенерована у Visual Studio

При взаємодії зі зчитувачем для зчитування і зберігання параметрів радіомітки використовується структура LotusCardParamStruct. Нижче наведено код цієї структури.

```
//Структура параметров
[StructLayout(LayoutKind.Sequential, CharSet =
CharSet.Auto)] public struct LotusCardParamStruct
{
    [MarshalAs(UnmanagedType.I4)] public int nCardType;//
    Тип метки
    [MarshalAs(UnmanagedType.ByValArray, SizeConst = 8)]
    public byte[] arrCardNo;// поле для ID карты
}
```

```

        [MarshalAs (UnmanagedType.I4)]                public int
nCardSize;// Размер(Емкость) метки
[MarshalAs (UnmanagedType.ByValArray, SizeConst = 64)]                public
byte[] arrBuffer;// Буфер для записи
        [MarshalAs (UnmanagedType.I4)]
        public int nBufferSize;// Длина буфера
        [MarshalAs (UnmanagedType.ByValArray, SizeConst = 64)]
public byte[] arrKeys;// Буфер ключа
        [MarshalAs (UnmanagedType.I4)]                public int
nKeysSize;// Длина буфера ключа
[MarshalAs (UnmanagedType.ByValArray, SizeConst = 256)]                public
byte[] arrCosResultBuffer; //Буфер результата
        выполнения
                                [MarshalAs (UnmanagedType.I4)]
                                public int unCosReultBufferLength;
    }

```

Ця структура використовується в методах класу LotusCardDriver в як параметр при зверненні до методів з бібліотеки DLL

3.3 Тестування програмного забезпечення

У таблиці 3.8 наведено опис тестових сценаріїв.

Таблиця 3.8 – Опис тестів

Приклад використання	Дія	Очікувані результати	Результатів
1. Запустіть програму	Запускає додаток	З'явиться основний додаток, поля даних порожні	Відповідають очікуваним результатам (рисунок Б.1)
2. Зчитування даних з тега	Користувач натискає на кнопку «Прочитати дані»	Поля заповнюються даними, які зчитуються з мітки	Досягніть очікуваних результатів (рисунок Б.2)

Продовження таблиці 3.8

Приклад використання	Дія	Очікувані результати	Результатів
3. Зчитування даних з порожньої етикетки	Користувач натискає на кнопку «Прочитати дані», мітка не містить даних	Відповідний. Повідомлення	Досягніть очікуваних результатів (рисунок Б.3)
4. Зчитування неправильних даних з бирки – неправильна дата введення в експлуатацію	Користувач натискає на кнопку «Прочитати дані», на етикетці вказана неправильна дата введення в експлуатацію	Відповідний. Повідомлення	Відповідають очікуваним результатам (рис. Б.4)
5. Зчитування неправильних даних з бирки – неправильна дата останньої перевірки	Користувач натискає на кнопку «Прочитати дані», на етикетці вказана неправильна дата останньої перевірки	Відповідний. Повідомлення	Відповідають очікуваним результатам (рисунок Б.5)
6. Зчитування неправильних даних з тега – неправильне значення зчитування	Користувач натискає на кнопку «Прочитати дані», етикетка містить неправильне значення зчитування	Відповідний. Повідомлення	Досягніть очікуваних результатів (рисунок Б.6)
7. Запишіть дані даних на порожню мітку	Користувач натискає на кнопку «Записати дані», мітка не містить даних	Відповідний. Повідомлення	Відповідають очікуваним результатам (рисунок Б.7)
8. Спроба записати неправильні дані на бирку – неправильна дата введення в експлуатацію	Користувач натискає на кнопку «Записати дані», невірно вказана дата введення в експлуатацію	Відповідний. повідомлення, дані не записуються в мітку	Досягніть очікуваних результатів (рисунок Б.8)

Продовження таблиці 3.8

Приклад використання	Дія	Очікувані результати	Результатів
9. Спроба записати неправильні дані на етикетку - неправильна дата останньої перевірки	Користувач натискає на кнопку «Зберегти дані», невірно вказана дата останньої перевірки	Відповідний. повідомлення, дані не записуються в мітку	Досягніть очікуваних результатів (рисунок Б.9)
10. Спроба записати неправильні дані на етикетку – неправильне значення зчитування	Користувач натискає на кнопку «Записати дані», вводиться неправильне значення зчитування	З'явиться повідомлення Відповідно, дані не записуються на етикетку	Відповідають очікуваним результатам (рисунок Б.10)
11. Перевірка діалогового вікна довідки	Користувач натискає на кнопку довідки «і»	З'явиться вікно з описом інструкції з використання програмного забезпечення	Відповідають очікуваним результатам (рис. Б.11)

Тести були написані для перевірки методів перевірки. Для тестування використовувався фреймворк NUnit. Нижче наведено тестовий код для перевірки роботи методу перевірки для пов'язаного параметра.

Перевірка показань за відповідним параметром

[Test]

```
public void VerificationOfReadingsByBoundedParameterTest()
```

```
{
```

```
    var item = new Item
```

```
    {
```

```
        Id = "1888429585",
```

```
        Manufacturer = "Bosh",
```

```
        Name = "ElectricDevice",
```

```
        StartDate = Convert.ToDateTime("2012-0615T00:00:00"),
```

```

        LastCheckDate = Convert.ToDateTime("2014-02-
11T00:00:00"),
        Readings = 585.0
    };
    var listOfItems = new ListOfItems();        listOfItems.Add(item); var
newItem = new Item
    {
        Id = "1888429585",
        Manufacturer = "Bosh",
        Name = "ElectricDevice",
        StartDate = Convert.ToDateTime("2012-0615T00:00:00"),
        LastCheckDate = Convert.ToDateTime("2014-02-
11T00:00:00"),
        Readings = 200
    };
    var verificationResult = new
    Verificator(listOfItems).VerifyReadingsByBoundedParameter(newItem)
    ;
    Assert.IsFalse(verificationResult);    }

```

(менше , ніж вимірювалося раніше), перевіряється метод класу `VerifyReadingsByBoundedParameter` з класу `Verificator`. Метод повинен повертати `false`, а об'єкт з цим значенням зчитування не пройде етап перевірки. Результати тестування представлені у зручному вигляді за допомогою фреймворку `JUnit`. Результати випробувань для методів верифікації показані на рисунку 3.7.

Таким чином, за допомогою наведених вище тест-кейсів було перевірено роботу функцій системи. Робота методів перевірки значень була перевірена за допомогою модульних тестів, які я написав. За результатами тестування можна зробити висновок, що програмне забезпечення виконує необхідні функції.

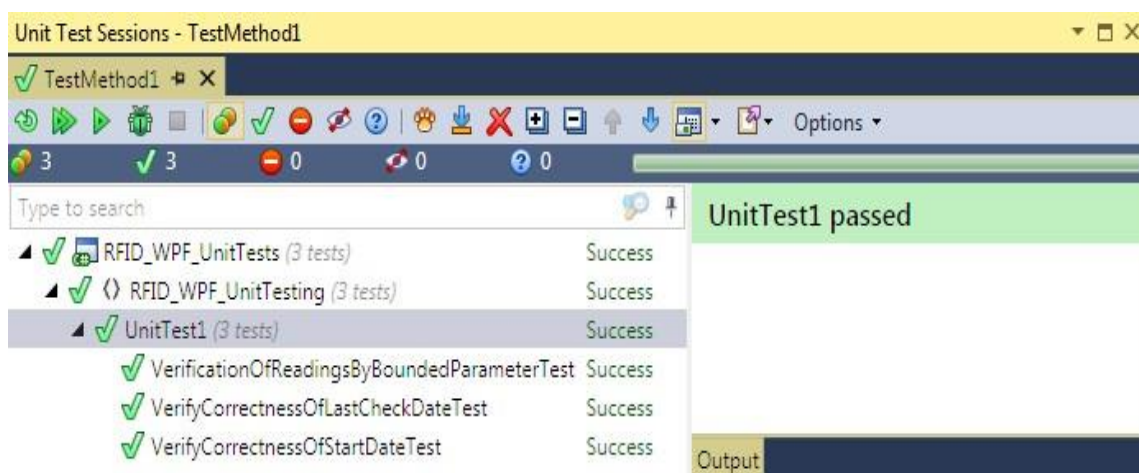


Рисунок 3.7 – Результати випробувань для методів

3.4 Аналіз якості коду

Метрики використовувалися для перевірки якості коду. Використовувані метрики розглянуті нижче.

Індекс супроводжуваності (Maintainability Index) — це комплексний показник якості коду. Цикломатична складність – показує структурну складність коду, тобто кількість різних гілок у коді. Чим вище цей показник, тим більше тестів потрібно написати, щоб повністю покрити код. Глибина успадкування. Цей показник показує для кожного класу, що він являє собою в ланцюжку успадкування. Наприклад, якщо є 3 класи А, В, С, В, успадковані від А, і С, успадковані від В, то значення цієї метрики для класів А, В і С дорівнюватиме 1, 2 і 3 відповідно.

Зв'язок класів – показує ступінь залежності класів один від одного. Враховуються унікальні класи з параметрів, локальних змінних, типу, що повертається, базового класу та атрибутів.

Хороший дизайн програмного забезпечення передбачає невелику кількість пов'язаних класів. Чим їх більше, тим складніше повторно використовувати цей клас в подальшому, а також підтримувати його, так як залежностей дуже багато.

Значення метрик коду, отриманих за допомогою вбудованих засобів у Microsoft Visual Studio, наведені на рисунку 3.8.

Code Metrics Results					
Filter: Cyclomatic Complexity		Min: 0	Max: <Maximum Value>		
Hierarchy		Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling
RFID_WPF (Debug)		86	180	10	55
{ } RFID_WPF		86	180	10	55
App		100	1	3	1
CLotusCardDriver		91	46	1	2
Item		87	28	1	10
ListOfItems		72	9	2	17
LotusCardParamStruct		100	0	1	0
MainWindow		75	47	10	28
OnLotusCardExtendReadWriteCallBack		100	4	1	2
Reader		67	40	1	14
Verificator		81	5	1	6

Рисунок 3.8– Значення отриманих метрик коду

Значення метрики MaintainabilityIndex = 86 вказує на те, що код простий в обслуговуванні. Значення метрики CyclomaticComplexity дорівнює 180, що вказує на структурну складність коду, цей показник може бути зменшений, якщо виключити деякі гілки.

За винятком класу, що представляє логіку головного вікна програми (MainWindow), всі класи мають значення метрики

DepthOfInheritance знаходиться від 1 до 3, що означає менш складний код і менший відсоток помилок у коді. Значення можливості ClassCoupling вказує на те, що багато класів мають велику кількість залежностей від інших класів, і кількість залежностей має бути зменшена.

3.5 Експериментальне дослідження програмних засобів

Для перевірки максимальної дальності зчитування був проведений експеримент: на кожен інтервал відстані від зчитувача зчитувалося по 50 міток, починаючи від 1 до 10 см. Максимальна заявлена дальність зчитування становила 10 см.

Для кожного з експериментів відсоток успішного читання розраховувався за формулою:

$$P = N_{\text{успех}} / N_{\text{общ.}}$$

где $N_{\text{успех}}$ — кількість гарно зчитаних міток; $N_{\text{общ.}}$ — общее кількість міток.

На рисунку 3.9 Представлено графік залежності успішного зчитування міток від відстані зчитування.

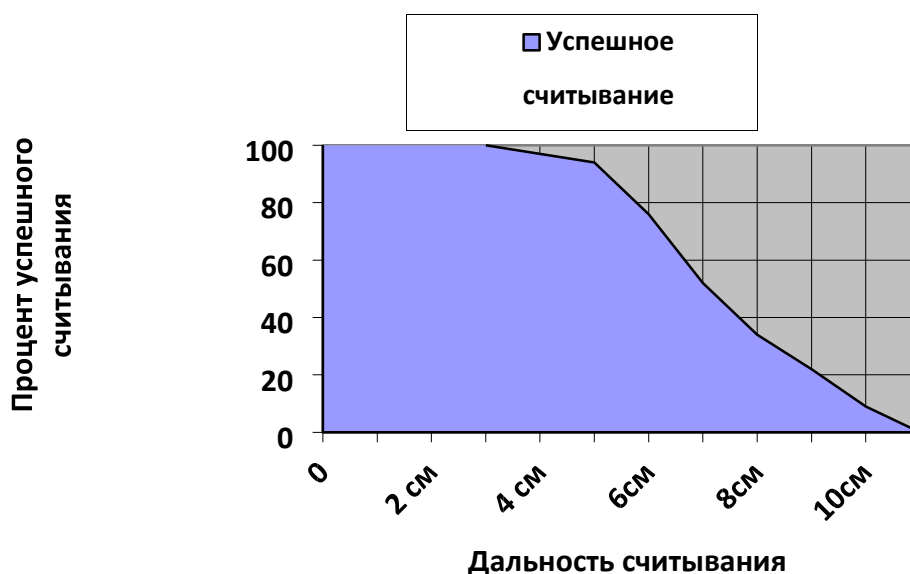


Рисунок 3.9 – Залежність успішного читання від відстані до читача

Горизонтальная вісь відображає відстань від мітки до зчитувача, а вертикальна вісь — відсоток успішного зчитування для експериментів, які зчитують дані з мітки на певній відстані. На рисунку 3.10 представлено зображення зчитувача, який використовується при розробці програмного забезпечення.



Рисунок 3.10 – Зображення читача

На підставі проведеного дослідження можна зробити висновок, що даний тип зчитувачів має діапазон зчитування менше заявленого виробником (заявлений діапазон зчитування в 2 рази більше фактичного). Теги, які використовувалися в експериментальних дослідженнях і розробці програмного забезпечення, показані на рисунку 3.11.



Рисунок 3.11 – Радіомітки, що використовуються при розробці програмного забезпечення

3.6 Висновки до розділу 3

Третій розділ присвячений розробці, тестуванню та аналізу якості коду програмних засобів збору даних про зчитування інформації.

Були реалізовані наступні завдання:

1. Оцінено трудомісткість розробки програмного забезпечення Коштів;
2. Архітектура системи збору даних про Дані працівників;
3. Проведено аналіз вимог до програмного забезпечення;
4. Розроблено варіанти використання;

5. Описано сценарії використання кейсів.
6. Програмні засоби для збору даних працівників;
7. Проведено ручне тестування розробленого програмного забезпечення
Коштів;
8. Для перевірки роботи програмного забезпечення розроблені модульні
тести
Коштів;
9. Проведено аналіз якості коду на основі метрик;
10. Завдання на удосконалення розробленого програмного забезпечення
Коштів; – проведено експериментальне дослідження роботи програмних
засобів.

ВИСНОВКИ

Під час виконання дипломної кваліфікаційної роботи були отримані наступні результати.

Проведено аналіз предметної області, існуючих завдань у сфері збору даних про зчитування інформації та енергоспоживання з карток працівників

Проведено огляд технології RFID та її застосування. Обґрунтовано можливість використання даної технології в досліджуваній предметній області.

Предметна область вивчається за допомогою методу онтологічного аналізу.

Побудовано метаонтологію предметної області. Потім була побудована спрощена онтологічна модель. На основі створеної моделі може бути побудована архітектура розроблених програмних засобів.

Розроблено архітектуру системи збору даних про дані працівників заводу або підприємства.

Проведено аналіз вимог до програмного забезпечення, впроваджено програмні засоби збору даних, проведено тестування розробленого програмного забезпечення.

Проведено аналіз якості коду на основі метрик, окреслено завдання щодо вдосконалення розробленого програмного забезпечення. Проведено експериментальне дослідження роботи програмних засобів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ізбродін М.Р. Використання технології радіочастотної ідентифікації в системах обліку енергії. / М.Р. Ізбродін, Л.В. Гур'янов // сб. XIII Междунар. Научно-техническая конференция. «Нові інформаційні технології та системи»–: Видавництво БП, 2016. – 404 р. 124-125.
2. Принцип роботи RFID-технології та її застосування // RTL Service
3. [Електронний ресурс]. Режим доступу:
https://rtlservice.com/ru/company/blog/princip_raboty_tehnologii_rfid_i_ee_primenenie/
4. Історія RFID-технології // ID Expert [Електронний ресурс]. URL:
http://www.idexpert. /reviews/935/?sphrase_id=39239 (дата звернення:
5. Технологія RFID // RST Invent [Електронний ресурс]. Режим доступу:
<http://www.rstinvent. /about/technology>
6. Капанський, А.А. Моделювання електроспоживання в технологічній системі водопостачання. / А.А. Капанський // Вестник Гомельського государственного технологического университета ім. П.О. Сухого. Випуск 2, 2016, с. 77-78.
11. Зв'язок ближнього радіусу дії: NFC EEPROM для витратомірів і систем