Contents

[Вступ 2](#_Toc23207411)

[Розробка функціоналу комплексу 4](#_Toc23207412)

[Огляд існуючих рішень 6](#_Toc23207413)

[Радіочастотна ідентифікація. Інформаційні системи з використанням радіочастотної ідентифікації 10](#_Toc23207414)

[Мітки для RFID 13](#_Toc23207415)

[Electronic Product Code (EPC) 13](#_Toc23207416)

[**Tag Antennas** 14](#_Toc23207417)

[Зчитувачі RFID 15](#_Toc23207418)

[Антенни зчитувачів 15](#_Toc23207419)

[Програмне забезпечення для передачі даних від RFID 17](#_Toc23207420)

# ВСТУП

Кожна людина так чи інакше займається веденням домашнього господарства. Воно включає в себе великий обсяг обов’язків, у тому числы ты, що можуть бути автоматизовані. Це допоможе людині зекономити час та, можливо, гроші. Наразі з’являється все більше і більше різних пристроїв, що націлені на автоматизацію побутової діяльності людини. Узагальнено це називається розумний дім. Розумний дім включає в себе управління та автоматизацію світлом, кліматом, розважальними системами та іншими побутовими приладами.

Одне з найважливіших питань в побутовій діяльності людини – запезпечення себе продовольчими та непродовольчими товарами. Існує повсякденна потреба у приготуванні їжі, купівлі ліків, побутової хімії тощо. В решті налічується дуже велика кількість найменувань товарів, що потребують постійної уваги людини під час здійснення покупок. Адже прикро повернутися додому і виявити, що якийсь необхідний товар не був куплений. Для запобігання подібних ситуацій деякі люди вирішують складати списки покупок, записуючи його на аркуші паперу або створюючи замітку в смартфоні. Таким чином, здійснювати покупки можна ефективніше: не витрачається зайвий час і кошти на купівлю непотрібних товарів. Якщо кожен раз чітко дотримуватися складеного списку покупок – не буде виникати проблеми нестачі якогось товару в домі.

Але при складанні списку покупок постає наступна проблема. Оскільки людина не тримає в голові де і в якій кількості знаходяться ті чи інші товари, їй доводиться кожного разу здійснювати моніторинг наявних товарів, і вже виходячи з нього – складати список покупок. Це монотонна робота, яку необхідно здійснювати кожного разу, коли людина збирається щось придбати. Отже це є витрата часу людиною.

Наразі серед пристроїв розумного дому немає такого, що вирішував би повністю описані проблеми. Існують програмні реалізації списку покупок що представляють собою зручний інтерфейс для формування цього списку за допомогою смартфону або іншого пристрою [?]. Але це програмне забезпечення надає лише базовий функціонал для роботи зі списком покупок: додавання товарів в список, ранжування за категоріями або видалення вже куплених товарів зі списку. Проблема подібних рішень у тому, що кожен раз необхідно заново складати список, так як не передбачено можливості відстежити, які товари вже були використані і мають потребу в повторному придбанні.

Також існує пристрій на основі сканеру штрих-кодів, яким необхідно зчитати штих-код на упаковці перед тим, як утилізувати товар. Після сканування штрих-коду цей пристрій автоматично додає ідентифікований товар до списку покупок. Але цей пристрій складає список тільки за тими товарами що були проскановані та викинуті. Він нічого не знає про вже існуючі товари у житлі, тобто може надавати хибну інформацію щодо необхідності купівлі того чи іншого товару. Наприклад, якщо людина придбала десять одиниць товару про запас, використала та викинула одну упаковку, зчитавши сканером штрих-код, то цей товар буде доданий до списку покупок. Це може не підходити для людини, оскільки в неї залишилось ще дев'ять одиниць. У такому випадку виявляється проблема відсутності повноти інформації у пристрою автоматизації про поточний стан наявних товарів у помешканні людини.

Сфера автоматизації та управління направлена на зменшення необхідності втручання людини у процеси[?]. Описана проблема може бути вирішена сучасними засобами інформаційних технологій. Програмно-апаратне рішення, що пропонується, має функціональність додавання всіх товарів що купуються людиною. Таким чином всі додані товари утворюють базу даних наявних речей у помешканні. Фіксація викинутих товарів створена на базі сканеру штрих-кодів та удосконалена застосуванням радіочастотної ідентифікації. У такий спосіб людині набагато зручніше взаємодіяти із системою: не потрібно сканувати штрих-код сканером. Радіочастотна ідентифікація здійснюється в той час, коли упаковка з радіо-міткою знаходиться на відстані кількох сантиметрів від пристрою. Саме така відстань може бути досягнута, якщо пристрій зчитування розмістити на внутрішній частині кошику для сміття, а у сміттєвий кошик викидати упаковку звичайним чином.

Об’єктом розробки магістерської дисертації є програмно-апаратний комплекс автоматизації ведення домашнього господарства з використанням радіочастотної ідентифікації.

Метою магістерської дисертації є впровадження автоматизації в процес ведення домашнього господарства людиною для зменшення витрати часу на організацію списку покупок та створення системи постійного моніторингу наявних товарів у помешканні людини.

Предметом розробки є підсистема апаратної ідентифікації предметів та веб застосунок для олбіку та управління предметами що наявні в домашньому господарстві.

Для досягнення поставленої мети були сформульовані та вирішені наступні задачі:

* формування вимог до функціоналу комплексу;
* огляд існуючих систем для автоматизації господарської діяльності;
* розробка структури комплексу;
* збір інформації про радіочастотну ідентифікацію;
* розробка підсистеми радіочастотної ідентифікації;
* огляд та вибір сканеру штрих-кодів;
* побудова апаратної підсистеми та розробка програмного забезпечення;
* аналіз розробленого апаратно-програмного комплексу.

# Аналіз предметної області автоматизації дому

В сфері автоматизації та сфері інформаційних технологій наразі існує велика кількість окремих інструментів для виконання тих чи інших задач. Для автоматизації побутової діяльності та створення продукту для розумного дому є кілька необхідних компонентів системи. В цьому розділі буде розглянуто складові для побудови системи автоматизації розумного дому, їх характеристики та порівняння. Також буде розглянуто існуючі рішення, що максимально наближено відповідають поставленій меті магістерської дисертації.

## 2.1 Складові системи автоматизації дому

Перше та найголовніше в системах автоматизації дому – пристрої що будуть взаємодіяти з навколишнім середовищем. Тобто безпосередньо сукупність датчиків та інших вимірювальних пристроїв що можуть переносити інформацію з навколишнього фізичного середовища до наступної ланки системи автоматизації.

Оскільки отриману інформацію з датчиків треба передавати на наступний рівень системи – постає друга проблема: засіб комунікації між ланцюгами системи. Це може бути як дротовий так і бездротовий зв’язок. Кожен з них налічує багато стандартів що широко розповсюджені та підтримуються різними пристроями.

Інформацію, отриману за допомогою датчиків, потрібно проаналізувати, обробити та виконати дії, що безпосередньо являють собою заміну людської діяльності. Це може бути реакція на зміни параметрів навколишнього середовища: зворотній зв’язок до виконавчих пристроїв або сигналізація користувачу про досягнення граничних значень параметрів.

Для взаємодії з боку людини, як користувача системи автоматизації, необхідно мати інструмент що відображав би інформацію та надавати змогу отримати налаштування від користувача. Це необхідно для того щоб система була гнучкою – могла змінювати базові критерії або граничні значення свої параметрів. Також користувач має можливість здійснити ручне регулювання системи для досягнення бажаного результата.

Для розробки програмно-апаратного комплексу автоматизації побутової діяльності людини було проаналізовано існуючі технології що можуть бути застосовані для кожного складового вузла.

## 2.2 Ідентифікація товарів

Для пристроїв що складають собою підсистему взаємодії з навколишнім середовищем необхідно чітко зчитувати необхідні параметри та передавати їх для подальшої обробки.

В запропонованому комплексі для пристроїв, що взаємодіють з навколищнім середовищем, поставлена задача ідентифікації товарів.

В сучасному світі для виконання таких задача існує технологія автоматичної ідентифікації та збору даних (Automatic Identification and Data Capturing - AIDC). Під цим терміном маються на увазі методи автоматичної ідентифікації об’єктів, збору даних про них та передача їх напряму до комп’ютерних систем без участі людини. Ідентифікатор об’єкту складається з інформації з якою можна встановити асоціацію за цим об’єктом. Ця інформація може бути представлена у вигляді зображення, звуку та навіть біометричних параметрів людини.

Задача автоматичної ідентифікації стоїть в тому, щоб отримати дані зовні шляхом обробки зображення або електромагніних хвиль. Для збору даних використовується перетворювач, що конвертує отриману інформацію у вигляді картинки або аудіозапису у електричні сигнали, що піддаються подільшій обробці комп'ютером. Комп'ютер порівнює інформацію на вході з існуючою базою даних або самостійно проводить ідентифікацію об'єкту.

Отримання даних може бути здійснено різними методами. Нижче наведено існуючі технології автоматичної ідентифікації об'єктів та збору даних:

* Штрих-код
* Радіочастотна ідентифікація
* Біометрика
* Магнітна стрічка
* Оптичне розпізнавання символів
* Смарт-картки
* Розпізнавання голосу
* Позиціонування в режимі реального часу(Real-time Locating Systems - RTLS)

З наведених технологій для ідентифікації продовльчих та нерподовольчих товарів, використовують розпізнавання за допомогою штрих-коду, радіочастотну ідентифікацію та оптичне розпізнавання символів.

Оптичне розпізнавання символів частіше застосовується при роботі з документами, але існує також практика ідентифікації товару за зображенням упковки. Такий спосіб дуже зручний оскільки для цього користувачу потрібна лише камера під’єднана до обчислювальної системи що будет обробляти отримане зображення. Але проблема постає в тому що для коректної роботи необхідно мати потужну нейронну модель, яка була б натренована на величезній кількості фотографій різноманітних товарів. Враховуючи динамічність ринку роздрібної торгівлі, появи нових товарів та ребрендингу або зміни дизайну упаковок вже існуючих, модель потрібно постійно оновлювати. Це призводить до великих затрат часу на створення, оновлення датасету зображень товарів та підтримання моделі в актуальному стані.

Ідентифікація продукту - це досить широка категорія маркування, що включає в себе такі функції, як відстеження товару, захист торгової марки та спосіб відображення інформації про вироблений товар. З бізнес-середовищем, що постійно змінюється, наповнюється контрабандним та фальсифікатом на популярні товари, з логістичними схемами постійно існуючими загрозами крадіжок та підміни товарів, маркування ідентифікації товару є критично важливим як для виробників, так і для споживачів.

Переважна більшість продовльчих та непродовольчих товарів що виробляються підприємствами для роздрібної торгівлі маркується згідно з міжнародними стандартами.

### 2.2.1 Ідентифікація за штрих-кодом

Штрих-код був винайдений в 1949 році аспірантами Інституту Технологій Дрекселя Бернардом Сілвером і Норманом Вудлендом [?]. Вони отримали патент на свій винахід у 1952 році. Однак штрихкоди не використовувались

в роздрібній торгівлі до 1967 року, коли коли сканери штрих-кодів почали поширюватися в продуктових магазинах у Сполучених Штатах Америки. Принцип кодування складається в тому, щоб цифри і букви представляють собою смуги різної ширини. Сьогодні більшість штрих-кодів своєю формою представляють прямокутники, оригінальна версія була представлена концентричними колами різної товщини.

На сьогоднішній день штрих-коди здобули широке застосування у різних сферах та контекстах. В першу чергу пересічній людині вони відомі за маркуванням на товарах в магазинах. Оскільки це прискорює та полегшує внутрішній процес доставки товару від виробника через магазини до покупця. Але це не єдине їх застосування. Наприклад, членські картки роздрібної торгівлі теж використовую штрих-коди для ідентифікаціїї клієнтів, що здійснюють покупку у конкретному магазині. Такий підхід дозволяє здійснювати індівідуальний маркетинг та краще розуміти модель покупок людини з точки зору магазину.

Переважна більшість існуючих способів відстеження товарів базується на ідентифікаціїї з використанням штрих-кодів. Наприклад, для відстеження штрих-кодами маркують арендовані автомобілі, багаж в аеропортах, відходи на атомних електростанціях , поштових вдправлень тощо.

Квитки на різні розважальні заходи (концерти, виставки, спортивні змагання) маркуються штрих-кодами. Людина може роздрукувати їх на принтері або завантажити на смартфон з якого ідентифікатор буде зчитаний на вході персоналом закладу.

В медичній сфері штрих-коди мають широке застосування. Їх використовують як для ідентифікації пацієнта, так і для ведення історії хвороби пацієнта (фіксація кожного відвідування у лікаря у цифровому форматі). Навіть виписка медикаментів може бути підкріплена штрих-кодом та зчитана на касі в аптеці для відображення необхідного списку ліків.

Основна передумова штрих-кодів полягає в тому, що буквено-числова інформація може кодуватися в смуги і пробіли різної ширини, надруковані на упаковці продукту. Саме за цими штрих-кодами ідентифікуються товари на складах в магазинах та безпосередньо на касі під час покупки кінцевим споживачем.

Штрих-коди використовують одновимірну схему кодування. Висота смуг забезпечує стійкість коду. Це означає, якщо частина символу пошкоджена, то все одно послідовність буде розпізнана коректно. Така характеристика є надзвичайно важливою, оскільки саме упаковка товару найчастіше зазнає пошкоджень.

Код містить у собі інформацію про виробника, категорію, до якої відноситься товар та безпеосередньо номер товару. Але в ньому не міститься інформація про ціну, дату виробництва товару тощо. Тож це лише ідентифікатор за яким вся необхідна інформація має бути отримана з інших джерел даних. Це може бути, наприклад, база даних магазину або складу виробника.

#### 2.2.1.1 Різновиди штрих-кодів

Існує багато різних видів кодування штрих-кодів. Всі вони поділяються на два великих класи: 1D та 2D штрих-коди. 1D – це саме ті види штрих-кодів, що наносяться в магазинах на упаковки та етикетки. Ще їх називають одновимірними штрих-кодами, тому що інформація міститься лише в одній площині – ширині смуг та пробілів. Висота смуг не відіграє ніякої ролі носія інформації. Тільки забезпечує стійкість коду. Саме тому, чим більша довжина повідомлення, тим більша довжина самого штрих-коду. Одновимірні коди добре підходять коли необхідно передати невеликий об’єм інформації, наприклад ідентифікаційний код товару за яким вже буде отримуватися інформація з бази даних де зберігається вся корисна інформація. Найпоширеніший стандарт одновимірних штрих-кодів в Європейських країнах – EAN-13 – представлений на рисунку 1. Слід зазначити, що часто декодована інформація розміщується під закодованим штрих-кодом для можливості ручного вводу людиною. Наприклад, якщо упаковку товар сильно пошкоджено, то касир має можливість ввести цифровий декодований надпис в касову систему вручну.

A picture containing object

Description automatically generated

Рисунок 1 – Приклад повідомлення “123456789012” у вигляді штрих-коду стандарту EAN-13.

2D штрих-коди це покращена технологія одновимірних кодів. Їх ще називають матричними або двовимірними штрих-кодами. Основна відмінність полягає в тому що інформація розміщається в двох вимірах – і по вертикалі і по горизонталі. Графічно такі коди виглядають набором впорядкованих спеціальним чином точок, квадратів, кіл, шестикутників тощо. Прямокутний 2D штрих-код може містити тисячі символів. Це рішення підходить коли необхідно розмістити досить велику кількість інформацію в компактному форматі. Таке застосування можна зустріти в логістичних структурах, де в код можна помістити адресу, контакти та ім’я отримувача та відправника посилки.

Найпоширенішим стандартом матричних штрих-кодів є QR Code. Через свою популярність він став підтримуватися стандартними застосуваннями до камер смартфонів, щоб кожна людина мала можливість зчитати такий код, та отримати інформацію з нього або навіть одразу взаємодіяти з нею. Так, наприклад, якщо закодувати адресу веб сайту в QR Code, то, зчитавши його смартфоном, можна одразу перейти за декодованим посиланням. Або, якщо в послідовності, зображеній на рисунку 2, закодувати SSID Wi-Fi мережі та пароль до неї, зчитавши такий код, смартфон буде намагатися до неї підключитися.

WIFI:S:<SSID>;T:<WPA|WEP|>;P:<password>;H:<true|false|>;

Рисунок 2 – формат запису інформації в QR Code для підключення до Wi-Fi мережі одразу після сканування смартфоном.

На рисунку 3 зображений приклад QR Code, що містить необхідну інформацію для доступу до Wi-Fi мережі з SSID «Test\_WiFi» та паролем «32168421».



Рисунок 3 – QR Code з інформацією для підключення до Wi-Fi мережі.

Окрім об’єму інформації, що може бути закодована в двовимірних штри-кодах, вони мають суттєву перевагу в порівнянні з одновимірними. Матричні штрих-коди можуть виправляти помилки, що дозволяє коректно декодувати інформацію навіть якщо графічний малюнок було трохи пошкоджено. Зокрема, QR Code має вбудовану корекцію помилок за кодами Ріда-Соломона, що зазвичай застосовується при роботі в контролерах оперативної пам’яті комп’ютера, записі та читанні інформації з оптичних дисків.

#### 2.2.1.2 Сканери штрих-коду

Для отримання інформації з штрих-коду існують спеціальні пристрої – сканери штриз-кодів. Це оптичний сканер, що, з надрукованого штрих-коду, декодує корисні дані та передає на комп'ютер або інший пристрій обробки даних. Пристрій сканеру складається з джерела світла, лінзи та оптичних сенсорів що перетворюють оптичні імпульси в електричні сигнали. Сканери штрих-кодів мають вбудовану інтегральну схему-декодер, що аналізує зображення баркода з сенсорів та передає отримані дані на вихідний порт.

Існує велика кількість реалізацій сканерів штрих-кодів. Кожен з них має своє місце та застосування у тій чи іншій сфері. Наприклад сканери можуть бути стаціонарними та переносними. Стаціонарні зручніше та доцільніше використовувати в касових апаратах, а переносні – на складах робітниками, що мають пересуватися по складу від одного товару, що необхідно сканувати, до іншого. На сьогоднішній день за технологіями, що застосовуються в сканерах, їх можна поділити на такі групи:

* Сканер-ручка
* Лазерний сканер
* Світлодіодний сканер
* Сканер на камери
* Всенаправлений сканер

Сканер-ручка представляє собою ручку з вбудованим джерелом світла та фотодіодом. Для зчитування штрих-коду таким типом сканеру, необхідно провести ручкою через всі полоси штрих-коду з незмінною швидкістю. В процесі сканування фотодіод вимірює інтенсивність відображеного світла, вираховує проміжок між наступною та попередньою зміною інтенсивності, таким чином визначаючи довжину смуг та відстань між ними. Чорні смуги поглинають світло, а білі – відбивають його, тож фотодіод на основі інтенсивності світла формує хвилю за різним рівнем напруги. Отримана хвиля і є відображенням штриз-коду в формі електричних імпульсів. Перевагою такої технології є її компактність. Але, процес сканування не є незручним, якщо мова йде про великі об'єми штрих-кодів що потрібно відсканувати. Оскільки людині необхідно плавно рукою проводити по поверхні надрукованого штрих-коду для коректного його розпізнавання.

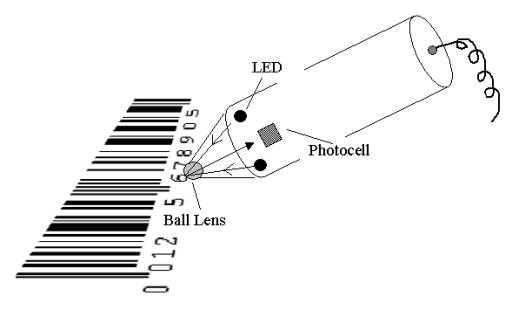


Рисунок – Штрихкод-сканер типу сканер-ручка

Лазрерні сканери праціюють майже так само, як і сканери-ручки за тією відмінністю, що в низ використовується лазерний промінь як джерело світла. Також лазерні сканери використвоють дзеркало або призму для заломлення променів після їх відбиття від штрих-коду щоб направити знову на поверхню, що необхідно зчитати. Так само, як і в сканері-ручці, для аналізу інтенсивності відбитого світла використовується фотодіод. В обох цих типах сканерів світло, що випромінюється сканерами швидко змінює свою яскравість в залежності від зчитуваних смуг, а використання фотодіодів для розпізнавання коду за інтенсивністю світла підходить тільки для заздалегідь відомої модуляції сигналу.

Світлодіодні сканери (також відомі як CCD або LED сканери), використовуються масиви сотень крихітних сенсорів, розташованих в рядок наверху сканера. Кожен сенсор вимірює інтенсивність світла прямо перед собою. Кожен окремий сенсор має дуже малі розміри, але через те, що вони розташовані в один ряд, вихідний електричний сигнал формується шляхом послідовного складання напруги на кожному сенсорі. Важливою відмінністю світлодіодних сканерів є те, що вони вимірюють інтенсивність світла навколо білих та чорних смуг штрих-коду. В той час як попередньо розглянуті технології вимірюють тільки відбите світло визначеної частоти, що генерує сам сканер.

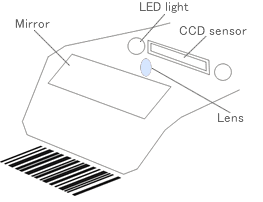


Рисунок – структура світлодіодного штрих-код сканеру

Сканери, що базуються на камерах, мабуть, є найсучаснішим доступним способом зчитувати штрих-коди пересічним людям. В першу чергу, адаптація технології скануваня штрих-кодів дозволила зчитувати двовимірні штрих-коди, такі як QR Code або Data Matrix, за допомогою камери смартфону.

Серед сканерів, що базуються на камерах, слід відокремити сканери, що використовують промислови камери з високою роздільною здатністю для того щоб захватити у кадр та зчитати кілька штрих-кодів одночасно. Всі коди, що потрапляють в поле зору камери декодуються миттєво. Для цього використовується запатентована технологія ImageID.

Багатонаправлені сканери штриз-кодів використовують послідовність прямих або кривих ліній для сканування в різних напрямках у формі зіркообразної Фігури Ліссажу так, щоб пересікати всі лінії штрих-коду. Майже всі такі лазери використовують лазер як джерело світла. Навідміну від простіших лінійних сканерів, багатонаправлені сканери можуть зчитати штрих-код під будь-яким кутом. Найчастіше такі сканери використовуються в касах супермаркетів, де упаковки скануються через скляне або сапфірове віконце. Такі сканери набули популярності в комерції, та варіюються від тих, що сканують на відстані кількох сантиметрів, до промислових конвеєрних сканерів, що можуть зчитати інформацію з предмета на відстані кількох метрів. Саме така технологія зчитування штрих-кодів є найбільш стійкою до пошкоджених кодів, що включає в себе як погану якість друку, так і пом’ятість або навіть розриви на поверхні штрих-коду.

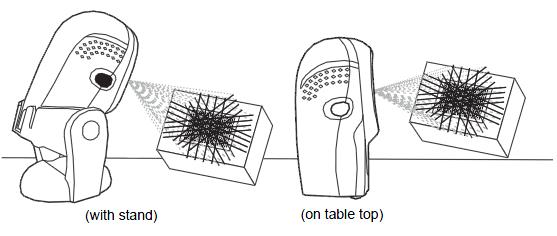


Рисунок – розсіювання променів багатонаправленого сканеру Motorola LS9208

2.2.2 Радіочастотна ідентифікація

Радіочастотна ідентифікація (RFID) - це форма бездротового зв'язку, яка використовує радіохвилі для ідентифікації та відстеження об'єктів. RFID – це загальний термін, що охоплює технології ідентифікації з різними стандартами. В їх число входять NFC та RAIN – дві технології, що є найпоширенішими серед усіх іниших засобів радіочастотної ідентифікації.

RAIN RFID – це бездротова технологія пассивної системи радіочастотної ідентифікації (без джерела енергії). Ця технологія використовується для додавання фізичних об'єктів в інформаційну систему. Це може бути інформаційна система що використовується в різних галузях, таких як інвентаризація, безпека, автентифікація, відстеження. RAIN наразі є найбільш розвиваючимся сегментом RFID та використовує єдиний міжнародний стандарт UHF Gen 2 (ISO/IEC 18000-63). За допомогою RAIN сьогодні до різних інформаційних систем підключено понад 20 мільярдів речей. Користувачі отримують інформацію в реальному часі про повсякденні речі, такі як одяг, медичні товари, деталі автомобілей, продукти харчування та інше.

RAIN RFID оцифровує світ фізичиних речей, тим самим дозволяє їм становитися частиною будь-якої інформаційної системи яка може відстежувати та реагувати на те, що відбуваєється з речами, отримувати та передавати корисні дані. Можна виділити наступні можливості, що відкриваються завдяки використанню радіочастотної ідентифікації:

* Унікальна ідентифікація одного предмета спроміж інших предметів того ж самого типу
* Ідентифікація місцезнаходження предмету без прямого вимірювання
* Ідентифікація багатьох предметів одночасно (до 1000 предметів за секунду)
* Визначення мысцезнаходження предмету на відстані від кільеох сантиметрів до кількох метрів

Радіочастотна ідентифікація надає можливість захищати та сильно впливати на життя користувачів, і в той же час здійснювати революцію у веденні бізнесу. Це зручне, недороге рішення, що дозволяє застосувати технологію радіочастотнох ідентифікації у широкому спектрі галузей – від рощдрібнох торгівлі до медицини, виробництва, розваг . Від управління та контролю за ланцюгом поставок до розпізнавання підробних ліків.

Можна зробити висновок, що радіочастотна ідентифікація допомагає:

* Автоматизувати інвентаризацію та відстеження товарів у сфері охорони здоров’я, виробництві, торгівлі та іншого бизнесу;
* Ідентифікація походження продукту, що дозволяє виявити дефективні або небезпечні одиниці, наприклад зіпсовану їжу, пошкодженні іграшку, ліки з строком зі сплинувшим строко придатності та ін.
* Перешкоджає потраплянню вищезазначених дефетиних товарів далі по ланцюгу постачальників, що упереджує отримання товару неналежної якості кінцевим споживачем;
* Підвищує якість обслуговування клієнтів, оскільки легше буде моніторити товари на складах, повертати бракований товар та ін.
* Дозволяє відстежувати ланцюг поставок, запезпечуючи більш ефективниу поставку за менші витрати бізнесу.
* Зменшує ризик збитків через крадіжку або некорректного обліку товарів
* Бездротове блоквваня, розблокання та конфігурація електричних пристроїв.
* Дозволяю налаштувани контроль доступу до певних зон, приміщень або пристроїв.

Система з використанням радіочастотної ідентифікації складається з трьох основних елементів:

* Предмета, до якого закріплена RFID мітка що унікально ідентифікує цей предмет.
* Пристрою, що забезпечує бездротовий двоаправлений зв’язок між предметами, описанними у першому пункті.

Програмного забезпечення, що збирає та трансформує дані з передавачів, надаючи інформацію в реальному часі до програмного забезпечення рівнем вище.

В базовій RFID системі мітки кріпляться до всіх речей, що мають тим чи іншим чином відстежуватися в інформаційній системі. RFID мітка складається з чіпа (інтегральнох схеми) та антенни, що надрукована, зтравлена та відштампована або нанесена парою на кріплення, що частіше за все є паперовим субстратом або Поліетилентерефталатом (ПЕТ). Поєднання чіпа та антенни вперсовується в паперову основу що з однієї сторони покривається клейкою основою або пакується у корпус (як, наприклад, мітки від домофону). Готові мітки доступні в різних варіаціях форм, розмірів включаючи стікери, картки пропуску та мітки на палети для промислової техніки. Сучасні розробки в сфері RFID роблять можливим ідентифікацію рідин та металів.

## 2.2.2.1 Мітки RFID

Мітки RFID однозначно ідентифікують предмет, на якому вони закріплені та мають у собі пам’ять що дозволяє розширити функціональність цього предмету. Чіп мітки живиться від радіохвиль, що поглинаються антенною та перетворюються у енергію живлення. Таким чином мітки не потребують власного живлення та читаються на відстані до 9 метрів по прямій лінії без перешкод.

В кжону мітку запрограмовано її власний ідентифікатор (tag identifier – TID) – це унікальний серійний номер, що записується у мітку виробником. Також у мітки може бути банк пам’яті для зберігання унікального ідентифікатора для відстеження товару, на який згодом буде прикріпленна мітк. Він зветься electronic product code або EPC.

Electronic Product Code (EPC)

Electronic Product Code (EPC) зберігаеється в пам’яті чіпа та займає зазвичай 96 біт даних. Схематичне зображення комірок можна побачити на рисунку 1. Перші 8 біт це заголовок, що ідентифікує версію протоколу. Наступні 28 біт ідентифікують організацію, що контролює цю мітку (номер організації за GS1). Наступні 24 біт – це ідентифікатор классу до якого належить продукт. Останні 38 біт це унікальний серійний номер самох мітки. Останні два поля заповнюються організацією, що виготовила цю мітку. Весь код уцілому може бути використаний як ключ до бази даних що ідентифікує предмет, до якого прикріплена ця мітка.

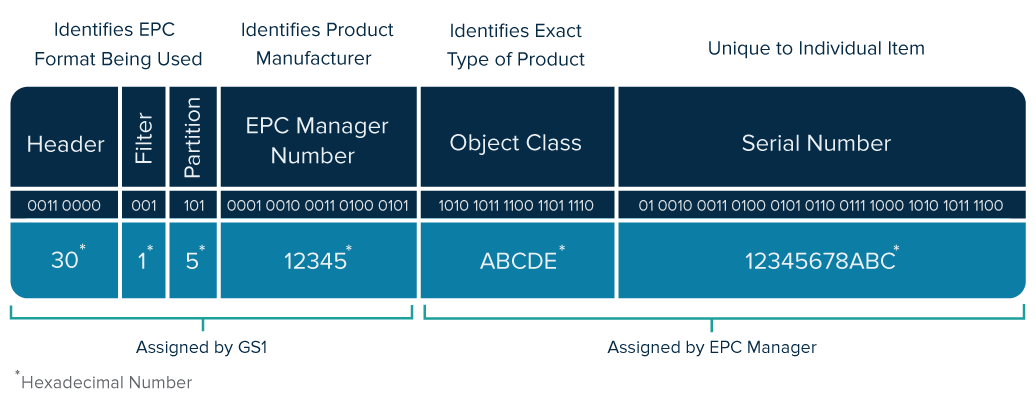


Рисунок 1 – Зміст Electronic Product Code (EPC)

### 2.2.2.2 Антена RFID мітки

Антенна мітки поглинає енергію радіохвилі і направляє її до чіпа. Тобто, чим більше площина антенни може поглинути більше енергії або ловити хвилі на більшій віддаленісті від джерела радіоховиль.

Не існує універсальної мітки для всіх способів застосування. Саме сферу застосування у більшості випадків визначає антенна мітки. Деякі мітки повинні працювати тільки на визначеному діапазоні частот, в той же час інші мають видавати найкращу потужність коли прикріплені до речей, що не пристосовані до бездротової комунцікації (наприклад, рідини та метали). Антенни можуть бути вироблені з різних матеріалів. Воні можуть бути надруковані, зтравлені, відштамповані чорнилами що проводять струм або навіть бути прикріалені парою до папіру.

### 2.2.2.3 Зчитувачі RFID

Зчитувачі RFID це пристрої що живлять та обмінюються інформацією за домпомогою бездротового зв'язку з мітками та передають дані від них до програмного забезпечення. Ці пристрої підтримують двонаправлений зв’язок з пристроями, на які прикріплені мітки, в радіусі їх допустимої дії. Зчитувачі можуть виконвати велику кількість завдань включаючи просту безперервну інвентаризацію, філтрування (пошук мітки за заданими критеріями), запис даних у певні мітки тощо.

Пристрох зчитування RFID міток можуть идентифікувати та визначити місц знаходження до 1000 міток в секунду. Зчитувачі можуть бути стаціонарними або мобільними та використовувати рінтегровану антенну для отримання даних від міток. Чіпи зчитувачів можуть бути вбудовані в такі пристрої як ручні зчитувачі, розумні автомати, пристрої для відстеження товарів, мобільні пристрох та ін.

Стаціонарні зчитувачі повинні мати антенну що посилає енергію через раідохвилі та дані з командами до міток. Оскільки ці зчитувачі часто виористовуються для автоматизації, вони можуть підтримувати додаткове підключення до зовнішніх сенсорів або до світлових пристроїв для сповіщення користувачі про завершення зчитування. Зазвичай такі пристрої підключені до хоста або до мережі щоб передавати дані від міток до додатків вищого рівня.

### 2.2.2.4 Антенни зчитувачів

Антени зчитувачів перетворюють електричний струм в едектромагнітні хвілі що потім випромінюються у ефір де вони вже будуть ловитися антенами на мітках та перетворюватися у зворотньому напрямку в електричний струм. Так само як і в антенах міток, існує велика кількість антен для зчитувачів та саме вона має відповідати вимогам до специфіки системі у якій планується застосувати радіочастотну ідентифікацію.

Два найбільш розповсюджених типи антен це антени з лінійною поляризацією та з круговою поляризацією (зображено на рисунку 2).

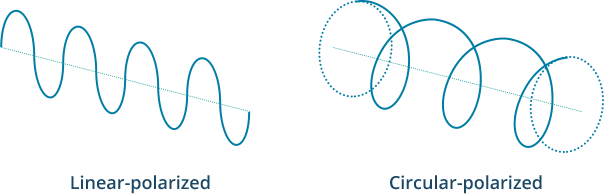


Рисунок 2 – Сигнали різних типів антен

Антени що випромінюють лінійні електричні поля, мають великий діапазон та високий рівень потужності який дозволяє сигналу проходити крізь різні матеріали щоб комунікувати з мітками. Але лінійні антени чутливі до росположення міток. Залежно від кута або розтаушвання мітки, зчитувачі з лінійно-поляризаційними антенами можуть краще або гірше тримати зв’язок з міткою.

Вибір антени також визначається відстанню між RFID зчитувачем та міткою, яку необхідно считати. Антена зчитувача можуть праціювати або в ближньому полі (короткий діапазон хвилі) або в дальному полі (довгий діапазон хвилі). У системах з коротким діапазоном хвилі зчитування мітки здійснюється на відстані менше 30см та використовується магнітний зв’язок для передачі енергії. Також в системах ближнього поля на якість зв’язку не впливає наявність у полі діелектриків таких як вода або метал.

У зчитувачах за антенами дального поля відстань між тегом та зчитувачем перевищує 30см і навіть може досягати кількох десятків метрів. Антени такого типу використовують електромагнітний зв’язок. Таким чином діелектрики можуть погіршити якість зв’язку між рідером та міткою.

### 2.2.2.5 Програмне забезпечення для передачі даних від RFID

В більшості випадків RFID зчитувачі використовують стандартизовану мову що зветься Low Level Reader Protocol або LLRP. Програмне забезпечення, розташоване на апаратному компоненті системи, називається прошивкою. Прошивка контролює роботу пристрою і, як правило, ініціалізує зв’язок із зовнішніми пристроями, такими як песональний комп’ютер.

Програмне забезпечення відправляє команди керування до зчитувача, який передае сигнали до RFID мітки та отримує через нього у відподвідь дані від мітки, надаючи можливість збирати та аналізувати ці данні у вищих шарах бізнес логіки системи.

## 2.3 Мережеві протоколи

Комунікація між пристроями займає дуже важливе місце в розробці систем автоматизації, що стосуються розумного дому. Мережеві технологіі надають можливість електричним пристроям здійснювати обмін даними між собою, а також із програмним забезпеченням, хмарними службами тощо. Інтернет побудований на стандартизованих протоколах, що забезпечують надійну та безпечну комунікацію між неоднорідними пристроями. Стандартні протоколи визначають правила, та вимоги до пристроїв для встановлення з’єдання, та успішної передачі даних в конкретній мережі.

В сфері автоматизації існує багато конкуруючих мережевих технологій. Одразу кілька технологій, розроблених різними корпораціями, націлені на різні ринки та сфери використання, такі як автоматизація дому, охорона здоров’я або промисловий інтернет речей. Але часто ці технології пропонують лише альтернативні варіанти реалізаціїї тих самих стандартних потоколів. Наприклад, протокол IEEE 802.15.4 описує роботу бездротових персональних мереж з низькою пропускною здатністю (LR-WPAN) і реалізується декількома конкуруючими технологіями, включаючи ZigBee, Z-Wave, EnOcean, SNAP та 6LoWPAN.

Технології, що використовуються для підключення до Інтернету, як, наприклад, Ethernet, часто можна застосовувати і для автоматизації. Однак нові технології розробляються спеціально для вирішення проблем пов’язаних з розвиненням інтернету речей. Якщо подивитись на стек технологій фізичної передачі, випливають наступні проблеми, шо постають перед побудовою мережі для інтернету речей:

* Безпека
* Маштабованість
* Мобільність
* Енергоефективність
* Пропускна здатність
* Надійність

Одним з важливих параметрів мережі – топологія. Для універсальної мережі інтернету речей найпоширенішими являються зіркова та сітчаста топологія. У зірковій топології кожен пристрій безпосередньо підключений до центрального вузла (шлюзу), що передає дані з підключених пристроїв на настунпий рівень. У сітчастих топологіях пристрої підключаються між собою в межах видимого діапазону. Кожен вузол в мережі може діяти як маршрутизатор, або навіть як шлюз. Сітчасті мережі більш складні, ніж мережі зірковою топологією, але мають перевагу в тому, що вони стійкіші до відмов, оскільки вони не покладаються на єдиний центральний шлюз.

Мережеві протоколи нижніх рівнів включають в себе як стыльниковий зв’яхок, Wi-Fi та Ethernet, так і більш спеціалізовані рішення, такі як LPWAN, Bluetooth Low Energy (BLE), ZigBee, NFC та RFID.

Важливою технологією комунікації на короткій дистанції є Bluetooth, що став дуже важливим для обчислень та є дуже поширеним в товарах для споживачів на ринку електроніки. Він вже давно використовується в мобільних телефонах, ноутбуках, різних носимих пристроїв, як смарт-годинники, фітнес-браслети тощо. Тож логічно, що він буде мати успіх на ринку інтернету речей. Новий Bluetooth Low-Energy (BLE) або Bluetooth Smart – важливий протокол для систем інтернету речей. Він забезпечує таку ж швидкість та діапазон дії, як і звичайний Bluetooth, але значно зменшує споживання електроенергії пристроями. Однак BLE насправді не призначений для передачі файлів, він більше підходить для невеликих шматочків даних. Але ця тезнологія має головну перевагу перед багатьма конкуруючими технологіями, враховуючи широку інтеграцію в смартфони та інші мобільні пристрої. Пристрої, що підтримують поточний стандарт Bluetooth 5.0 можуть використовувати швидкість передачі даних до 2 Мбіт / с, що вдвічі більше, ніж підтримує Bluetooth 4.2. Пристрої також можуть спілкуватися на відстані до 240 метрів, що в чотири рази перевищує аналогічний показник попереднього стандарту версії Bluetooth 4.2 [https://www.bluetooth.com/specifications/bluetooth-core-specification/bluetooth5]. Слід зазначити, що всі ці показники є дійсними і для BLE версії протоколу, що надає розумним пристроям ще й додаткову енергоефективність.

ZigBee, як і Bluetooth, має велику розповсюженість, але традиційно більше в промислових умовах. ZigBee PRO та ZigBee Remote Control (RF4CE) базуються на протоколі IEEE802.15.4, який є стандартною технологією бездротових мереж, що працює на частоті 2,4 ГГц. Цей протокол використовується в умовах відносно рідкого обміну даними при низькій пропускній здатності (250Кбіт/с) на обмеженій території (в межах 100 м) в порівнянні за BLE. ZigBee RF4CE має деякі значні переваги в складних системах забезпечуючі нижче енергоспоживанням, високу безпеку, надійність та високу масштабованість з великою кількістю вузлів. Це дає можливість використовувати бездротові мережі в автоматизації та управлінні там, де існує велика кількість сенсорів що комунікують мід собою (M2M – Machine to machine). Остання версія ZigBee - це нещодавно запущений 3.0, який по суті є об'єднанням різних бездротових стандартів ZigBee в єдиний стандарт.

Z-Wave - це технологія радіочастотного зв'язку з низькою потужністю, яка в першу чергу призначена для автоматизації дому. Зокерма, для таких пристроїв, як контролери ламп та датчики температури, вологи тощо. Оптимізований для надійного зв'язку з малою хатримкою невеликих пакетів даних зі швидкістю передачі даних до 100 Кбіт/с, технологія працює в діапазоні до 1 ГГц і є стійкій до перешкод від інших бездротових технологій в діапазоні 2,4 ГГц, таких як WiFi, Bluetooth або ZigBee. Він підтримує топологію сітчастої мережу без необхідності вузла координатора і може масштабуватися до 232 пристроїв.

Підключення пристроїв інтернету речей до Wi-Fi часто є очевидним вибором для багатьох розробників, особливо зважаючи на поширеність WiFi в домашніх умовах. Усюди існує розвинена інфрастуктура для забезпеченя передачі даних з високою швидкістю та надійністю. В даний час найпоширенішим стандартом WiFi, який використовується в будинках та багатьох підприємствах, є 802.11n. Він пропонує велику пропускну здатність (100-300Мбіт/с), що добре підходить для передачі файлів, але може бути занадто затратним для багатьох сценаріїв використання інтернету речей.

Якщо система вимагає комунікації між пристроями на великій відстані - можна скористатися можливостями стільникового зв'язку GSM / 3G / 4G. Хоча стільниковий зв’язок явно здатний надсилати велику кількість даних, особливо для 4G, витрата, а також споживання електроенергії, буде занадто високою для багатьох випадків. Але це може ідеально підходить для проектів, що використовують тільки сенсори та потребують низьку пропускну здатністю даних, та будуть надсилати їх у невеликій кількості через Інтернет. Такоже існують нові технології, що розроблені спеціально для потреб комунікації пристроїв інтернету речей. Стандарти LPWAN NB-IoT та LTE-M спрямовані на забезпечення енергоефективних, дешевих варіантів зв'язку з використанням існуючих стільникових мереж. NB-IoT - це найновіший із цих стандартів, орієнтований на зв'язок на дальні відстані між великою кількістю пристроїв, розміщених у приміщенях. LTE-M та NB-IoT були розроблені спеціально для інтернету речей, однак, як було описано раніше, існуючі стільникові технології також часто застосовуються для бездротового зв'язку на великій відстані.

Широко розповсюджений для дротового підключення в локальних мережах, Ethernet реалізує стандарт IEEE 802.3. Не всі пристрої інтернету речей повинні бути бездротовими. Деякі з них можуть бути розміщені стаціонарно та бути запезпеченими проводними комінкаціями. Наприклад, сенсорні блоки, що встановлюються в системі автоматизації будівлі, можуть використовувати такі технології дротових мереж, як Ethernet. Це можуть бути датчики диму, вогню тощо.

Протокол NFC (Near Field Communication) використовується для зв'язку дуже малого діапазону (до 4 см), наприклад, для тримання картки NFC або мітки поруч із зчитувачем. NFC часто використовується для платіжних систем, але він також корисний для пропускних охоронних систем, наприклад, для авторизації доступу.

RFID розшифровується як радіочастотна ідентифікація. Мітки RFID містять в собі ідентифікатори та дані. Ці мітки кріпляться до пристроїв з метою подальшого зчитування RFID-сканером. Типовий діапазон RFID менше метра. Мітки RFID можуть бути активними або пасивними. Пасивні мітки ідеально підходять для пристроїв без акумуляторів, оскільки сканер під час зчитування живить мітку енергією радіохвиль, що їм випромінюються. Пасивна мітка надсилає за рахунок отриманої енергії відповідь, яку приймає сканер. Активні мітки періодично транслюють свій ідентифікатор у навколишній простір, тоді як допоміжні пасивні теги стають активними. Dash7 - це протокол зв'язку, що використовує активні RFID мітки. Він розроблений для використання в програмах промислового інтернету речей для безпечної комунікації на великій відстані. Як і в NFC, типовий випадок використання для RFID - це відстеження товарно-матеріальних цінностей у роздрібних та промислових додатка.

Дальність роботи мережі визначається відстанню, на якій дані зазвичай передаються між пристроями, приєднаними до мережі. Можна виділити наступні категорії мереж за дальністю роботи:

* Персональна мережа (Personal Area Network) - короткий діапазон, де відстані вимірюються в метрах. Наприклад, носимі гаджети для фітнесу, що обмінюються даними через BLE з додатком на смартфоні користувача.
* Локальна мережа (Local Area Network) - це короткий, до середнього дальності, діапазон, де відстань може бути до сотень метрів. Наприклад, домашня автоматизація або датчики, встановлені на виробничій лінії підприємства, що спілкуються по Wi-Fi із головним вузлом.
* Міська мережа (Metropolitan Area Network) – мережа, в якій дальність дії вимірюються до кількох кілометрів. Наприклад розумні датчики паркування, встановлені по всьому місту та з'єднані в мережу сітчастої топології.
* Широка мережа (Wide Area Network) – мережа, що покриває велику площу, де відстані можна виміряти в кілометрах. Наприклад, сільськогосподарські датчики, встановлені на великій фермі, що використовуються для моніторингу мікроклімату навколишнього середовища.

Мережа повинна бути розроблена для отримання даних з пристроїв до місця, де вони будуть аналізуватися та оброблятися. Отже, мережевий протокол, повинен відповідати діапазону використання. Наприклад, не слід вибирати BLE для програм, що повинна працювати на відстані декількох кілометрів. Якщо передача даних через необхідний діапазон представляє складність, необхідно розглянути крайові обчислення (edge computing). Така технологія передбачає аналіз та обробку даних на пристрої без необхідності передачі їх для подальшої обробки.

Пропускна здатність або обсяг даних, що можуть бути передані за певний проміжок часу, обмежує швидкість, з якою дані можна збирати з пристроїв та передавати на наступний рівень системи. На пропускну здатність впливає кілька факторів:

* Обсяг даних, який генерує кожен пристрій
* Кількість пристроїв, розгорнутих у мережі

Розмір пакету обраного мережевого протоколу повинен відповідати розміру даних, які зазвичай передаються. Неефективно надсилати пакети, заповнені порожніми даними. Але також з іншого боку можуть бути і накладні витрати для розділення великих фрагментів даних на занадто багато маленьких пакетів. Швидкість передачі даних не завжди симетрична. Це означає, швидкість передачі може бути меншою, ніж швидкість завантаження. Отже, якщо між пристроями існує двостороння комунікація, потрібно попередньо обрахувати передачу даних. Бездротова та стільникова мережі традиційно мають низьку пропускну здатність, тож необхідно чітко вирішити, чи є бездротова технологія правильним вибором для великих об’ємів даних в поточній ситуації.

Також необхідно визначити, чи потрібно передавати всі вихідні дані. Одним з варіантів може бути передача меншої кількості даних шляхом менш частого збору інформації, збору меншої кількості параметрів або проведення деякої фільтрації на пристрої, щоб скинути незначущі дані. Агрегування даних перед тим, як їх передачею зменшить обсяг даних, що підлягають передачі. Агрегація або подрібнення не завжди підходять для даних, що залежать від часу або затримки.

Передача даних з пристрою потребує витрат енергії. А для передачі даних на великі діапазони потрібна більша потужність, ніж на короткому діапазоні. Необхідно врахувати, що пристрій живиться від акумулятора. Щоб продовжити термін служби акумулятора, можна переводити пристрій у сплячий режим, коли він не працює. Рекомендується моделювати споживання енергії пристрою при різних навантаженнях та різних умовах навколишнього середовища, щоб впевнитися, що потужності джерела живлення вистачить для передачі необхідних даних, в умовах обраної мережевої технології.

Безпека мережі - це важливий параметр, тому краще вибрати мережеву технологію, що реалізує захист каналу від одного кінця до іншого, включаючи аутентифікацію, шифрування та захист відкритого порту. Наприклад, IEEE 802.15.4 включає модель безпеки, що включає контроль доступу, перевірку на цілісність повідомлення, конфіденційність повідомлення та захист від повторного відтворення.

Якщо використовується технологія Wi-Fi, можена забезпечити захист мережі за допомогою протоколу Wi-Fi Protected Access II(WPA2). Щоб забезпечити конфіденційність та цілісність даних для зв'язку між додатками, не зазвичай використвуються криптографічні протоколи Transport Layer Security (TLS) або Datagram Transport-Layer Security (DTLS), що базується на TLS, але адаптований для ненадійних з'єднань, що використовують UDP пакети. TLS шифрує дані програми та забезпечує їх цілісність.

При розробці архітектури мережі слід взяти до уваги захищеність портів. Такий підхід передбачає, що лише порти, необхідні для зв'язку зі шлюзом або додатками вищього рівня, залишаються відкритими для зовнішніх з'єднань. Усі інші порти повинні бути відключені або захищені брандмауерами.

Вибір мережевої технології для впровадження в систему автоматизації дому передбачає компромісні рішення. В подальшому, цей вибір вплине на параметри кінцевих пристроїв, а, в свою чергу, серед більшості з цих параметрів існує залежність. Наприклад, діапазон мережі, швидкість передачі даних та споживання енергії безпосередньо пов'язані між собою. Якщо збільшити діапазон мережі або швидкість і обсяг даних, що передаються, між пристроями, буде потрібна додаткова потужність для передачі даних та коректної роботи пристроїв.

Для базового проекту домашньої автоматизації критерій енергоефективності, має невелике значення. Оскільки пристрій, швидше за все, буде живитись безпосередньо від розетки. Обмеження пропускної здатності та нестабільний зв’язок були б важливішими параметрами, тому технологія WiFi може бути доцільним вибором, оскільки вона забезпечує розумну пропускну здатність, а також полегшує створення проекту за допомогою вже наявного обладнання в домі. Однак, Wi-Fi не оптимізований для пристроїв малої потужності, тому цей вибір може бути не найкращим вибором для пристроїв, що живляться від акумулятора.

## 2.4 Інструменти прототипування

### 2.4.1 Платформа Arduino

Arduino - це платформа прототипування з відкритим кодом, заснована на простому у використанні апаратному та програмному забезпеченні. Плати з мікроконтролером Arduino здатні зчитувати вхідні дані - світло, зафіксоване датчиком, натиснення кнопки або повідомлення у Twitter - та перетворювати їх на вихідний сигнал - активація двигуна, включення світлодіода, публікація інформації в Інтернеті. Ви можете сказати вашій платі, як працювати, надіславши набір інструкцій мікроконтролеру, який знаходиться на ній. Для цього необхідно використовувати мову програмування Arduino та програмне забезпечення Arduino Integrated Development Environment (IDE).

Протягом багатьох років Arduino була ядром тисяч проектів - від предметів побуту до складних наукових інструментів. Всесвітня спільнота виробників - студентів, любителів, художників, програмістів та професіоналів - зібралася навколо цієї платформи з відкритим кодом, їх внески додали до неймовірної кількості доступних знань, які можуть бути дуже корисними як для початківців, так і для експертів.

Arduino народилась в Інституті дизайну взаємодії Івреа як простий інструмент для швидкого складання прототипів, призначений для студентів, котрі не мали досвіду роботи з електронікою та програмуванням. Як тільки вона дійшла до більш широкої аудиторії, плата Arduino почала змінюватися, щоб адаптуватись до нових потреб, урізноманітнившись від простих 8-бітних плат до продуктів для IoT-додатків, носіїв, 3D-друку та вбудованих середовищ. Програмне та апаратне забезпечення Arduino є відкритими, що дає можливість користувачам самостійно збирати, програмувати плати і, зрештою, адаптувати їх під конкретні потреби.

Завдяки простому та доступному досвіду користування, Arduino використовувається у тисячах різних проектів та програм. Програмне забезпечення Arduino просте у використанні для новачків, але водночас досить гнучке для досвідчених користувачів. Воно працює на операційних системах Mac, Windows та Linux. Викладачі та студенти використовують Arduino для створення недорогих наукових інструментів, доведення принципів хімії та фізики або щоб отримати перший досвід програмування та робототехніки. Дизайнери та архітектори будують інтерактивні прототипи, музиканти та художники використовують його для інсталяцій та експериментують з новими музичними інструментами. Arduino – один з основних інструментів для вивчення нового. Будь-хто - діти, любителі, художники, програмісти - можуть почати програмувати плати, лише дотримуючись покрокових інструкцій набору або ділитися ідеями в Інтернеті з іншими членами спільноти Arduino.

Існує багато інших мікроконтролерів та платформ з мікроконтролерами, що використовуються для фізичних обчислень: Parallax Basic Stamp, BX-24 Netmedia, Phidgets, MIT’s Handyboard та багато інших пропонують подібний функціонал. Платформа Arduino також спрощує процес роботи з мікроконтролерами, але вона має певні переваги над іншими системами:

* Недорога - Плати Arduino відносно недорогі порівняно з іншими мікроконтролерними платформами. Найдешевшу версію Arduino можна зібрати вручну. Однак навіть попередньо зібрана Arduino коштує менше 50 доларів.
* Крос-платформена - Програмне забезпечення Arduino (IDE) працює на операційних системах Windows, Macintosh OSX та Linux. Більшість систем мікроконтролерів обмежені Windows.
* Просте, зрозуміле середовище програмування - Програмне забезпечення Arduino (IDE) є простим у використанні для початківців, але при цьому досить гнучким для професіоналів.
* Відкрите програмне забезпечення, що піддається розширенню - Програмне забезпечення Arduino є інструментом з відкритим кодом, доступним для розширення досвідченими програмістами. Мова може бути розширена за допомогою бібліотек C++, і ті, хто бажають зрозуміти технічні деталі, можуть перейти з Arduino на мову програмування AVR C, на якій він заснований. Так само можна додати код AVR C безпосередньо у програми Arduino.
* Відкрите апаратне забезпечення, що піддається розширенню - Плани плат Arduino публікуються під ліцензією Creative Commons, тому досвідчені конструктори схем можуть розробити власну версію пристрою, розширивши та вдосконаливши його. Навіть відносно недосвідчені користувачі можуть створити макет пристрою, щоб зрозуміти принцип його роботи.
* Проект базується на плані мікроконтролерної плати, розробленої кількома постачальниками. Ці системи пропонують набори цифрових та аналогових пінів вводу та виводу, які можуть поєднуватися з різними платами розширення (так званими шилдами) та іншими схемами. Плати оснащені послідовними інтерфейсами зв’язку, у тому числі USB на деяких моделях, для завантаження програм з персональних комп’ютерів. Для програмування мікроконтролерів проект Arduino забезпечує інтегроване середовище розробки на основі проекту Processing, що містить підтримку мов програмування C і C++.
* Перша Arduino була представлена у 2005 році як недорогий і простий спосіб створити пристрої, що взаємодіють зі своїм оточенням за допомогою датчиків і пускачів. У склад поширених прикладів таких пристроїв входять прості роботи, термостати та детектори руху.
* Плати Arduino випускаються комерційно у заздалегідь зібраному вигляді або як комплекти для самостійного збирання. Специфікації дизайну апаратного забезпечення знаходяться у відриктому доступі. Adafruit Industries підрахували, що у середині 2011 року було продано понад 300 000 офіційних Arduino, а в 2013 році - вже 700 000.

#### 2.4.1.1 Апаратне забезпечення

Плата Arduino історично складається з 8-, 16- або 32-розрядного мікроконтролера AVR (хоча з 2015 року використовуються мікроконтролери інших виробників) з додатковими компонентами, які полегшують програмування та включення в інші схеми. Важливим аспектом Arduino є його стандартні роз’єми, які дозволяють користувачам підключати плату процесора до різноманітних взаємозамінних модулів доповнення, відомих як шилди. Певні шилди взаємодіють з платою Arduino безпосередньо через піни, але більшість адресуються індивідуально через послідовну шину I²C - тому багато шилдів можна складати та використовувати паралельно.

До 2015 року офіційні Arduino використовували мікросхеми Atmel megaAVR, зокрема ATmega8, ATmega168, ATmega328, ATmega1280 та ATmega2560, а у 2015 році були додані мікросхеми інших виробників. Кілька інших процесорів також використовувалися на пристроях, сумісних з Arduino. Більшість плат містять лінійний регулятор напругою 5 В і кварцовий генератор 16 МГц (або керамічний резонатор у деяких варіантах), хоча певні конструкції, такі як LilyPad, працюють на частоті 8 МГц і обходяться без бортового регулятора напруги через конкретні обмеження форм-фактора. Мікроконтролер Arduino попередньо запрограмований завантажувачем, який спрощує завантаження програм у флеш-пам’ять на мікросхемі, порівняно з іншими пристроями, які зазвичай потребують зовнішній програматор. Це робить використання Arduino простішим, дозволяючи використовувати звичайний комп’ютер як програматор. Наразі завантажувач optiboot - це завантажувач за замовчуванням, встановлений на Arduino UNO.

На концептуальному рівні при використанні інтегрованого середовища розробки Arduino всі плати програмуються через послідовне з’єднання. Реалізація залежить від апаратної версії. Деякі послідовні плати Arduino містять схему зсуву рівня для перетворень між логічними рівнями RS-232 та сигналами рівня TTL. Сучасні плати Arduino програмуються за допомогою універсальної послідовної шини (з англ. “Universal Serial Bus” (USB)), реалізованої за допомогою чіпами адаптера USB-to-serial, таких як FTDI FT232. Деякі плати, наприклад, більш пізні моделі Uno, замінюють мікросхему FTDI окремим чіпом AVR, що містить USB-to-serial мікропрограму, яку можна перепрограмувати за допомогою її власного піна ICSP. Інші варіанти, такі як Arduino Mini і неофіційний Boarduino, використовують знімний USB-to-serial адаптер або кабель, Bluetooth або інші. Якщо замість Arduino IDE застовуються традиційні інструменти мікроконтролера, використовується стандартне програмування AVR ISP.

Плата Arduino дозволяє використовувати більшість контактів вводу-виводу мікроконтролера в інших схемах. Diecimila, Duemilanove та Uno передючають наявність 14 цифрових пінів вводу-виводу, 6 з яких можуть виробляти сигнали з широтно-імпульсною модуляцією, і 6 аналогових входів, які водоночас можуть використовуватися як 6 цифрових пінів вводу-виводу. Ці піни розташовані у верхній частині плати у вигляді гнізд з внутрішнім діаметром 2,5 мм. Плата Arduino Nano і сумісна з Arduino плата Bare Bones і Boarduino можуть містити піни-штекери на нижній стороні плати, які можна підключити до плати без паяння.

Є багато плат, сумісних з Arduino та похідних від Arduino. Деякі є функціонально еквівалентними Arduino і можуть використовуватися взаємозамінно. Багато хто вдосконалює базову Arduino, додаючи вихідні драйвери. Інші плати - електрично еквівалентні, але змінюють форм-фактор, зберігаючи сумісність із шилдами, однак ця сумісніть зберігається не завжди. Існують і варіанти, що використовують зовсім інші процесори з різним рівнем сумісності.

Оригінальне апаратне забзепечення Arduino було виготовлено італійською компанією Smart Projects. Деякі плати під маркою Arduino були розроблені американськими компаніями SparkFun Electronics і Adafruit Industries. На сьогоднішній день комерційно було вироблено сімнадцять версій обладнання Arduino.

Приклади плат Arduino:

* Arduino Diecimila in Stoicheia
* Arduino Duemilanove (rev 2009b)
* Arduino UNO
* Arduino Leonardo
* Arduino Mega
* Arduino MEGA 2560 R3 (front side)
* Arduino MEGA 2560 R3 (back side)
* Arduino Nano
* Arduino Due (ARM Cortex-M3 core)
* LilyPad Arduino (rev 2007)
* Arduino Yun

Arduino та Arduino-сумісні плати використовують друковані плати розширення, які називаються шилдами. Вони підключаються до конекторів Arduino. Шилди можуть забезпечувати управління двигуном, GPS, Ethernet, LCD або макетування (прототипування). Деякі шилди можна виготовити самостійно.

Приклади шилдів Arduino:

* Можна укладати кілька щитів
* Шилд з гвинтовою клемою у вигляді крила
* Adafruit Motor Shield з гвинтовими зажимами для підключення до двигунів
* Adafruit Datalogging Shield зі слотом для карт Secure Digital (SD) і чіпом реального часу (з англ. “real-time clock”)
* HackARobot Fabric Shield - розроблений для Arduino Nano для підключення двигунів і датчиків, таких як гіроскоп або GPS, та інших друкованих плат, таких як WiFi, Bluetooth, RF і т.д.

#### 2.4.1.2 Програмне забезпечення

Програми Arduino можуть бути написані на будь-якій мові програмування з компілятором, який генерує бінарний машинний код. Atmel надає середовище розробки для своїх мікроконтролерів, AVR Studio і новішу Atmel Studio.

Проект Arduino надає інтегроване середовище розробки Arduino (IDE), яке є кросплатформним додатком, написаним на Java. Воно виникло з IDE для проекту мови програмування Processing і проекту Wiring. Середовище розробки Arduino включає в себе редактор коду з такими функціями, як підсвічування синтаксису, зіставлення дужок і автоматичний відступ, а також надає простий механізм (один клік) для компіляції і завантаження програм на плату Arduino. Програма, написана за допомогою IDE для Arduino, називається “скетч”.

Arduino IDE підтримує мови програмування C і C++, використовуючи спеціальні правила організації коду. Arduino IDE містить програмну бібліотеку під назвою “Wiring” з проекту Wiring, яка надає можливість користування загальними процедурами введення і виведення. Типовий скетч Arduino C/C++ складається з двох функцій, які скомпільовані і пов’язані з програмю main() у виконуваній циклічній програмі:

* setup() - функція, яка запускається один раз при запуску програми і ініціалізує налаштування
* loop(): функція, яка викликається багаторазово, поки плата не вимкнеться

Після компіляції і зв’язування з набором інструментів GNU, що також міститься у дистрибутиві IDE, Arduino IDE використовує програму avrdude для перетворення коду в текстовий файл в шістнадцятковому коді, який завантажується в плату Arduino програмою-завантажувачем у мікропрограму плати.

2.4.2 Arduino Mega 2560

Arduino Mega - це мікроконтролерна плата на основі ATmega1280 (таблиця даних). Вона складається з 54 цифрових пінів вводу-виводу (з яких 14 можуть використовуватися як ШІМ-виходи), 16 аналогових входів, 4 UART (апаратних послідовних портів), кварцового генератора 16 МГц, USB-порта, роз’єму живлення, роз’єму ICSP, і кнопки скидання. Вона містить все необхідне для підтримки мікроконтролера; просто підключіть її до комп’ютера за допомогою USB-кабелю або увімкніть за допомогою AC-to-DC адаптером чи акумулятора, щоб почати роботу. Mega сумісний з більшістю шилдів, розроблених для Arduino Duemilanove або Diecimila.

|  |  |
| --- | --- |
| Мікроконтролер | ATmega1280 |
| Робоча напруга | 5 В |
| Вхідна напруга (рекомендовано) | 7-12 В |
| Вхідна напруга (межі) | 6-20 В |
| Цифрові піни вводу-виводу | 54 (з яких 15 забезпечують вихід ШІМ) |
| Аналогові вхідні піни | 16 |
| Постійний струм на піни вводу-виводу | 40 мA |
| Постійний струм на піни 3.3 В | 50 мA |
| Флеш-пам’ять | 128 Kб, з яких 4 Kб використовуються завантажувачем |
| SRAM | 8 Kб |
| EEPROM | 4 Kб |
| Тактова швидкість | 16 МГц |

Arduino Mega може заряджатись через USB-з’єднання або від зовнішнього джерела живлення. Джерело живлення вибирається автоматично.

Зовнішнє (не USB) живлення може подаватися або від AC-to-DC адаптера, або від акумулятора. Адаптер можна підключити, вставивши центральний позитивний штекер 2,1 мм в роз’єм живлення плати. Клеми від акумулятора можна вставити в піни Gnd і Vin роз’єму POWER.

Плата може працювати від зовнішнього джерела живлення від 6 до 20 В. Однак, якщо на нього подається напруга менше 7 В, на пін 5 В може подаватися напруга менше 5 В, і плата може бути нестабільною. При використанні напруги більше 12 В регулятор напруги може перегрітися і пошкодити плату. Рекомендований діапазон становить від 7 до 12 В.

Існують наступні піни живлення:

* VIN. Вхідна напруга на платі Arduino при використанні зовнішнього джерела живлення (на відміну від 5В від USB-підключення або іншого регульованого джерела живлення). Ви можете подавати напругу через цей пін або, якщо ви подаєте напругу через роз’єм живлення, отримати до нього доступ через цей пін.
* 5V. Регульоване джерело живлення, яке використовується для живлення мікроконтролера та інших компонентів на платі. Це може відбуватися або від VIN через вбудований регулятор, або від USB чи іншого регульованого джерела живлення 5В.
* 3V3. Джерело живлення 3,3В, що генерується вбудованим чіпом FTDI. Максимальний струм споживання становить 50 мА.
* GND. Піни заземлення.

ATmega1280 має 128 Кб флеш-пам’яті для зберігання коду (з них 4 Кб використовується для завантажувача), 8 Кб SRAM і 4 Кб EEPROM (які можна читати і записувати за допомогою бібліотеки EEPROM).

Кожен з 54 цифрових контактів на Mega може використовуватися як вхід або вихід за допомогою функцій pinMode(), digitalWrite() та digitalRead(). Вони працюють на 5В. Кожен пін може забезпечити або отримати максимум 40 мА і має внутрішній підтягуючий резистор (відключений за замовчуванням) 20-50 кОм. Крім того, деякі піни мають спеціалізовані функції:

* Serial: 0 (RX) та 1 (TX); Serial 1: 19 (RX) та 18 (TX); Serial 2: 17 (RX) та 16 (TX); Serial 3: 15 (RX) та 14 (TX). Використовується для отримання (RX) та передачі (TX) послідовних даних TTL. Піни 0 і 1 також підключені до відповідних пінів послідовної мікросхеми FTDI USB-TTL.
* Зовнішні переривання: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), та 21 (interrupt 2). Ці піни можуть бути налаштовані так, щоб викликати переривання на низькому значенні, при зростанні чи падінні фронта або при зміні значення (функція attachInterrupt ()).
* ШІМ: від 2 до 13 та від 44 до 46. Забезпечте 8-бітний вихід ШІМ за допомогою функції analogWrite().
* SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). Ці піни підтримують зв’язок SPI, який, хоча і забезпечується базовим обладнанням, наразі не входить до мови Arduino. Піни SPI також розбиті на пін ICSP, який фізично сумісний з Duemilanove та Diecimila.
* LED: 13. Є вбудований світлодіод, підключений до цифрового піна 13. Коли пін має значення HIGH, світлодіод увімкнено, коли значення піну - LOW, він вимикається.
* I2C: 20 (SDA) і 21 (SCL). Підтримка зв’язку I2C (TWI) за допомогою бібліотеки Wire. Ці піни знаходяться не у тому ж місці, що й піни I2C на Duemilanove або Diecimila.

Mega має 16 аналогових входів, кожен з яких забезпечує 10 біт роздільної здатності (тобто 1024 різних значень). За замовчуванням вони вимірюють від заземлення до 5 В. Верхню межу діапазону можна змінити за допомогою піна AREF та функції analogReference().

На платі є кілька інших пінів:

* AREF. Опорна напруга для аналогових входів. Використовується з analogReference().
* Reset. Встановіть значення LOW, щоб скинути мікроконтролер. Зазвичай використовується для додавання кнопки скидання шилдів, які блокують її на платі.

У Arduino Mega є низка засобів для взаємодії з комп’ютером, іншим Arduino або другими мікроконтролерами. ATmega1280 пропонує чотири апаратних UART для послідовного зв’язку TTL (5 В). FTDI FT232RL на платі транслює один із них через USB, а драйвери FTDI (входять до програмного забезпечення Arduino) забезпечують віртуальний COM порт до програмного забезпечення на комп’ютері. Програмне забезпечення Arduino містить послідовний монітор, який дозволяє надсилати прості текстові дані з та до плати Arduino. Світлодіоди RX і TX на платі будуть блимати, коли дані передаються через мікросхему FTDI та USB-з’єднання до комп’ютера (але не для послідовного зв’язку на пінах 0 і 1).

Бібліотека SoftwareSerial дозволяє здійснювати послідовний зв’язок на будь-яких цифрових пінах Mega.

ATmega1280 також підтримує зв’язок I2C (TWI) та SPI. Програмне забезпечення Arduino включає бібліотеку Wire для спрощення використання шини I2C.

Замість того, щоб вимагати фізичного натискання кнопки скидання перед завантаженням, Arduino Mega розроблений таким чином, що дозволяє скинути її за допомогою програмного забезпечення, що працює на підключеному комп’ютері. Одна з апаратних ліній управління потоком (DTR) FT232RL підключена до лінії скидання ATmega1280 через конденсатор 100 нанофарад. Коли ця лінія підтверджен (взята на низькому рівні), лінія скидання падає досить довго, щоб скинути мікросхему. Програмне забезпечення Arduino використовує цю можливість, щоб дозволити завантажувати код, просто натиснувши кнопку завантаження в середовищі Arduino. Це означає, що завантажувач може мати коротший час очікування, оскільки зниження DTR може бути добре узгоджено з початком завантаження.

Це налаштування має й інші наслідки. Коли Mega підключена до комп’ютера, на якому працює Mac OS X або Linux, вона скидається щоразу, коли з’єднання з нею здійснюється за допомогою програмного забезпечення (через USB). Упродовж наступних півсекунди завантажувач працює на Mega. Хоча він запрограмований на ігнорування неправильно сформованих даних (тобто будь-чого, окрім завантаження нового коду), він буде перехоплювати перші кілька байтів даних, що надсилаються на плату після відкриття з’єднання. Якщо скетч, запущений на платі, отримує одноразову конфігурацію чи інші дані під час першого запуску, переконайтеся, що програмне забезпечення, з яким він взаємодіє, чекає секунду після відкриття з’єднання та перед надсиланням цих даних.

У Arduino Mega є перезавантажуваний полізапобіжник, який захищає USB-порти вашого комп’ютера від коротких замикань і перенапруг. Хоча більшість комп’ютерів забезпечують власний внутрішній захист, запобіжник забезпечує додатковий рівень захисту. Якщо до USB-порту застосовано більше 500 мА, запобіжник автоматично перерве з’єднання, поки не буде усунено коротке замикання або перевантаження.

Максимальна довжина та ширина плати Mega - 10,16 та 5,33 см відповідно, роз’єм USB та роз’єм живлення виходять за її рамки. Три отвори для гвинтів дозволяють прикріпити плату до поверхні або корпусу. Зауважте, що відстань між цифровими пінами 7 та 8 становить 160 міл (4,064 мм), що не є кратним відстані у 100 міл (2,54 мм) між іншими пінами.

Mega розроблена так, щоб бути сумісною з більшістю шилдів, призначених для Diecimila або Duemilanove. Цифрові піни від 0 до 13 (а також суміжні піни AREF та GND), аналогові входи від 0 до 5, пін живлення та піни ICSP знаходяться в рівнозначних місцях. Крім того, основний UART (послідовний порт) розташований на тих же пінах (0 і 1), як і зовнішні переривання 0 і 1 (піни 2 і 3 відповідно). SPI доступний через піни ICSP як на Mega, так і на Duemilanove та Diecimila. I2C не розташований на тих же пінах на Mega (20 і 21), що і на Duemilanove та Diecimila (аналогові входи 4 і 5).

# 3 Огляд існуючих рішень

## 3.1 Hiku

Hiku - це розумний помічник, призначениий для створення списків покупок, оснащений сканером бар-кодів та функцією керування голосом.

Перш ніж скористатися Hiku, необхідно завантажити додаток-компаньйон на телефон (Hiku сумісний як із пристроями iOS, так і з Android) та створити обліковий запис. Для того щоб з’єднати додаток з Hiku, його потрібно розмістити на екрані свого смартфона таким чином, щоб об’єктив був спрямований вниз. Об’єктив Hiku - це прозоре вікно, розташоване в нижній частині пристрою. У разі виникнення проблем під час користування пристроєм, Hiku забезпечує довідку на сайті або в центрі підтримки користувачів. В основному допоміжні матеріали представлені у формі відеозаписів і містять наглядні інструкції до користування своїм продуктом.

Після спарювання пристроїв, можна почати використовувати Hiku для створення списків покупок. Варто зазначити, що Hiku підтримує 2,4 ГГц-мережі, а не 5 ГГц. Hiku підлаштовується під голос і може розуміти кілька мов, включаючи англійську, французьку, іспанську та інші. Говорити потрібно чітко та прямо у пристрій.

Під час сканування продуктів Hiku видає два звукові сигнали. Ці звуки означають, що сканування спрацювало та продукт був успішно доданий у ваш список покупок. Три звукові сигнали повідомляють про те, що штрих-код не було розпізнано. У цьому випадку можна просто сказати назву продукта і Hiku додасть його до списку. Цей продукт буде збережено у пам’яті та при наступному скануванні штрих-коду, товар буде розпізнано.

Hiku надає можливість підключення до Walmart.com та Peapod. Якщо у цих магазинах є товари зі списку, їх можна придбати напряму.

Hiku розроблений для iOS (10 або новішої версії) та Android, однак також працює з Apple Airport, швидкою бездротовою базовою станцією Apple. Підтримує 802.11b/g/n Wi-Fi, 2.4 ГГц. Hiku досить міцний, водостійкий. Оснащений бортовим мікрофоном, розпізнавання мови надане Nuance Communications, Inc. Hiku устаткований літій-іонним акумулятором. Для зарядки використовується мікро USB-порт.

Hiku оснащений модулем 1D лінійного сканування зображень. Він підтримує найпопулярніші символи, такі як UPC-A, Code 128, Code 29, UPC-E, EAN-8, EAN-13 та ін. Hiku має базу даних зі штрих-кодами з понад 17 мільйонами записів, більшість з яких призначені для американських продуктів. Однак Hiku можна використовувати в інших країнах світу.

Переваги Hiku:

* Простий у використанні.
* Є підтримка кількох списків.
* Режим очікування наднизької потужності, більше двох місяців автономної роботи при звичайному використанні
* Контроль голосом, підтримка кількох мов.
* Hiku не знімає додаткові кошти за збереження списків у хмарному сховищі.

Недоліками Hiku є наступні:

* Вартість Hiku вище середнього – 49 доларів.
* Неможливість додавати товари у кастомізований список за допомогою голосу (вони будуть додані у список за замовчуванням).
* Неможливість додавати товари у кастомізований список при першому скануванні (доданні) товару (вони будуть додані у список за замовчуванням).
* Неможливість ділитися списками поза додатком або з іншими зареєстрованими користувачами додатку Hiku.

## 3.2 GeniCan

GeniCan - це розумний пристрій для сміттєвих баків, який допомагає складати свій щотижневий продуктовий список, відстежуючи продукти, які ви використовували під час їх утилізації. Пристрій кріпиться до побутових сміттєвих баків та має зчитувач штрих-коду з бездротовим підключенням. Користувач сканує штрих-код товару, перш ніж викинути його, і GeniCan автоматично заповнює список покупок у додатку.

Якщо товар (наприклад, фрукт) не має штрих-коду, GeniCan починає “говорити”: запитує користувача, який товар потрібно додатти до списку покупок. Коли користувач відповідає, GeniCan використовує функцію розпізнавання голосу, щоб зрозуміти команду. Мова розпізнавання одна – англійська. Під час роботи з пристроєм, його стан можна ідентифікувати за кольором індикатора: синій – прокидається зі сну, зелений – готовий сканувати, червоний – готовий записувати аудіо. Успішне виконання операції завершується характерним сигналом. У разі невдачі GeniCan повідомить, що не зрозумів голосову команду.

GeniCan працює місяцями на наборі з 4 батарейок типу AA і використовує Wi-Fi радіо для підключення до маршрутизатора та надсилання даних в Інтернет. Ви можете налаштувати його для автоматичного замовлення товарів завдяки інтеграції з послугами Amazon Dash Replenishment Services. Компанія GeniCan заявила, що компанія планує в майбутньому додати інших онлайн-партнерів, крім Amazon.

Додаток GeniCan доступний для мобільних пристроїв iOS та Android. Списками, які зберігаються в додатку GeniCan, можна ділитися з іншими користувачами, які можуть додавати товару до списку або отримати доступ до нього під час покупок.

GeniCan підключається до мережі 802.11b/g/n Wi-Fi, 2.4 ГГц. На даний момент GeniCan зчитує коди UPC та EAN, поточна база даних складається з продуктів у США. Поза межами США, швидше за все, знайдуться товари, які не будуть ідентифіковані при першому скануванні. Додаток GeniCan дозволяє додавати описи та зображення для будь-яких продуктів, які не відображаються в поточній базі даних, тому при наступному скануванні товар буде розпізнаний.

Переваги GeniCan:

* Простий у використанні.
* Є підтримка кількох списків.
* Контроль голосом.
* Можливість ділитися списками поза додатком.

Недоліками GeniCan є наступні:

* Вартість GeniCan 149 доларів.
* База даних розрахована на США.
* Функція розпізнавання голосом підтримує тільки одну мову.

# 4 Розробка програмно-апаратного комплексу

В цьому розділі будуть описані етапи розробки програмно апаратного комплексу для автоматизації побутової діяльності людини. Спираючись на дослідження стану справ в цій галузі було вирішено створити власну реалізацію системи, що допомогає людині у повсякденному житті шляхом автоматизації визначення списку покупок.

## 4.1 Принцип роботи комплексу

Система базується на тому, що завжди має інформацію про те, які товари наявні в поточний момент часу в домі. Тобто, система відслідковує, коли які товари були куплені людиною, та коли вони були використані та вкинуті до сміттєвого кошика. В першу чергу це стосується продовольчих товарів. Оскільки саме вони мають найменший «життєвий цикл» у домашньому господарстві людини. Це за більшістю своєю є натуральні продукти, що мають певний термін придатності. Тож вони мають бути використані людиною, або утилізовані після того, як спливає їх термін придатності.

Важливим параметром цієї системи є придатність товарів до ідентифікації. В попередніх розділах було зазначено, що більшість товарів маркуються вирбониками за домопогою використання штрих-кодів. Такий спосіб є найдешевшим, оскільки вимагає лише надрукувати на упаковці штрих-код що має невелику площину та складається лише з чорних та білих смуг різної товщини. Існують інші види ідентифікації, але вони вже потребують використання додаткових пристроїв що підвищує ціну виробництва продукту. Але, оскільки світовий ринок торгівлі переважно орієнтований саме на роботу зі штрих-кодами, то доцільно було скористатися цим видом ідентифікації товару. Тож було вирішено, що одним з видів ідентифікації, що має підтримувати розроблювана система – є ідентифікація за штрих-кодом.

З точки зору зручності використання, ідентифікація за допомогою штрих-кодів має певні недоліки, а саме:

* Для зчитуваня штрих-коду необхідно мати прямий доступ до самого коду. Будь-яка перепона унеможливить коректне зчитування коду.
* Штрих-коди не мають пам’яті та можливості запису та зчитування додаткової інформації окрім безпосередньо ідентифікатора. Наприклад, штрих-код не може місти термін придатності товару, а містить лише ідентифікатор виробника та товару.
* Штрих-коди потребують людьского часу та роботи, оскільки кожен штрих-код треба сканувати індивідуально. Наприклад, наводити сканер на код, та утримувати його в такому положені доки сканер не сигналізує про успішне розпізнавання штрих-коду.
* Штрих-коди мають невисокий рівень безпеки, оскільки їх можна легко відтворити або підробити.
* Штрих-коди легко пошкоджуються, оскільки друкуються назовні упаковки товару та піддаються фізичному впливу навколишньої середи.

На сьогоднішній день, незважаючи на недоліки штрих-кодів, вони залишаються найпоширенішим засобом ідентифікації. Тому в розроблюваній системи було закладено підтримку такого типу ідентифікації. Але для більшого ступеню автоматизації процесу необхідно використовувати засіб ідентифікації, що не потребує людського втручання. Такою технологією була обрана радіочастотна ідентифікація – RFID.

Суттєвою перевагю з боку покращення автоматизації є те – що для використання такого способу автоматизації людині потрібно лише причепити RFID мітку для будь-якого фізичного предмета. З цього моменту предмет може бути доданий в систему на облік. Це робить можливим додавання навіть тих предметів, упаковки яких не містять штрих-коди. Зчитувачі RFID міток одразу спрацьовують, коли в їх полі досяжності потрапляє мітка. Для цього не має потреби спеціальним образом підставляти мітку під зчитувач. Достатньо просто її піднести, не менше ніж на визначену характерисиками зчитувача і мітки відстань.

У кожного товару є певний час існування від того моменту, як він був куплений в магазині, до того моменту, як він буде утилізований людиною після використання. Назвемо цей час існування життєвим циклом товару у домашньому господарстві людини. Як було зазначено, для того щоб визначити, які продукти мають бути куплені знов, необхідно відслідковувати, коли товар був використаний та утилізований. Інакше кажучи, система має виявляти, коли життєвий цикл цього товару закінчився. Для відслідковування життєвого циклу було вирішено спиратися на ідентифікатор, як і на етапі додавання товару у систему. Для цього на сміттєвому кошику, або неподалік від нього необхідно ідентифікувати товар. Ідентифікація товару біля сміттєвого кошика означає, що він вже використаний або більше не потрібен людині. Тобто його життєвий цикл скінчився. Виходячи з цього, можна зробити припущення, що цей товар необхідно відновити. Але це припущення є частково хибним. Оскільки не так багато товарів є такими, що постійно знаходяться у використанні. З іншого боку, в цілях збереження повноти забезпеченності житла всіма необхідними речами, варто припускати сценарій, коли людина хоче придбати новий товар на заміну старому, навіть якщо це не є часто використовуваний товар. Для набуття визначенності в цьому питанні необхідно індивідуально підходити до кожного потенційного користувача системи, досліджувати його модель поведінки з точки зору здійснюваних покупок.

Для отримання більшої інтеграції розроблюваної системи автоматизації з повсякденним життям людини є доцільним розглядати взаємодію із вже існуючими екосистемами розумного дому.

В контексті розроблюваного програмно-апаратного комплексу як системи моніторингу товарів у домі, впровадження технології радіочастотної ідентифікації систему дозволяє користувачам не здійснювати додаткових дій для того щоб надавати системі вхідні дані. Для того щоб система помітила товар як викинутий, користувачеві необхідно лише попередньо розмістити пристрій зчитування RFID міток неподалік від сміттєвого кошика. Коли людина буде викидати товар у кошик – RFID зчитувач виявить мітку в полі своєї дії, визначить її ідентифікатор і відправить дані на обробку до верхнього рівня системи. З точки зору людини такий процесс роботи системи ніяк не вплине на її дії в порівнянні з тим, як вона викидала товари у сміттєвий кошик до того як почала користуватися даною системою автоматизації ведення домашнього господарства. Слід відмітити, що, при використанні ідентифікації за допомогою штрих-коду, людина повинна змінити звичний алгоритм викидання у сміття. Оскільки вона повинна не забути відсканувати штрих-код на упаковці. Якщо під час відкриттся упаковки штрих-код був пошкоджений, то його буде складно успішно відсканувати під час викидання упаковки у сміття.

Однак використання радіочастотної ідентифікації призводить до додаткових витрат для користувача. Нажаль, в наш час маркування RFID мітками товарів в магазинах можна зустріти дуже рідко. На всі товари їх не вигідно чіпляти ні виробникам, ні магазинам. Такому маркуванню підлягають тільки цінні товари. Адже збитки з крадіжки таких товарів перевищили б вартість маркування RFID мітками. Тому користувачам програмно-апаратного комплексу необхідно самостійно придбати мітки на липкій основі, щоб всі куплені товари можна було самостійно маркувати. У світі існує тенденція до заміни штрих-кодів на RFID мітки. Вже в цьому році, третя по величині мережа роздрібної торговлі, після Wal-Mart та The Home Depot, Kroger представила новий формат магазинів, де замість штрих-кодів для обліку використовуються радіочастотні мітки. Спільно з Microsoft вони реалізували розрахунок на касі через мобільний додаток, використовуючі чіпи NFC [<https://progressivegrocer.com/new-grocery-technology-piloted-kroger-microsoft-shows-store-future>].

Такі події з участю всесвітньо відомих компаній в сфері маркетингу свідчать підтвердженням того, що у майбутньому світ може відмовитися від використання штрих-кодів на користь RFID міток. Тож можна сказати, що для сьогоднішніх потенційних користувачів розроблюваної системи незручності, пов’язані з необхідністю самостійного маркування товарів радіочастотним мітками, є лише тимчасовими. В подальшому, коли маркування RFID мітками буде забезпечуватися мережами роздрібної торгівлі, або навіть виробниками, користувачам не треба буде витрачати час та кошти на маркування товарів самостійно.

Для користувачів, що не можуть дозволити собі витрати на мітки як на додатковий ресурс, який потрібно постійно поновлювати для функціонування системи в поточному варіанті, залишається єдиним способом ідентифікувати товари – штрих-кодом. Для цього не має необхідності постійно купляти допоміжні пристрої. Сканер можна зчитати просто камерою смартфону.

Програмне забезпечення спроектовано таким чином, що може приймати інформацію з будь-якого з трьох видів пристроїв – сканеру RFID, сканеру штрих-кодів та смартфону. Це забезпечує досить широкий вибір альтернатив у контексті існуючих засобів ідентифікації товарів. Отриманий ідентифікатор по мережі Інтернет буде передано до серверу та опрацьовано програмним забезпеченням.

В розроблюваному програмно-апаратному комплексі програмна частина має відповідати за обробку інформації що надходить від фізичних пристроїв та представлення її у зручному для користувача вигляді. Найбільш універсальним варіантом реалізації є створення веб-застосунку. Оскільки користувачу веб-застосунок буде доступний усюди: з будь-якого мобільного пристрою, з комп’ютеру чи ноутбуку та навіть з вбудованої мультимедійної або пропрієтарної електроніки (як пульт керування розумним домом). Для ідентифікації користувача в веб-застосунку передбачено створення аккаунту. Такий спосіб зручний для використання всією родиною. Кожен член сім’ї може увійти в аккаунт та побачити, які продукти наявні у домі. Або ж, після повернення з магазину, легко додати нові товари.

Веб-застосунок, як спосіб взаємодії системи з людиною, є одним з найкращих, оскільки це вже звичний та знайомий кожному інструмент. Він надає повну свободу у тому де і як його використовувати. Необхідно лише мати доступ до Інтернету, що в наш час не є великою проблемою у містах.

Весь програмно-апаратний комплекс призначений для використання в приміщені, де необхідно впровадити автоматизацію ведення домашнього господарства. Апаратна частина комплексу складається з двох пристроїв. Один з них необхідно розмістити на кошику для сміття, оскільки він відповідає за фіксацію факту утилізації упаковки товару до сміття. Інший пристрій призначений для маркування товарів радіочастотними мітками. За його допомогою користувач фіксує в системі появу конкретної радіочастотної мітки, після чого зв’язує мітку з певним продуктом з тих, що вже наявні у системі моніторингу. Обидва пристрої є автономними одиницями оскільки мають власне джерело живлення - акумулятор. Його можна заряджати використовуючи звичний інтерфейс micro-USB. Такий мобильний підхід дозволить без зайвих зусиль розмістити ці два пристрої в будь-якому житлі без необхідності забезпечувати ці пристрої постійним енергопостачанням.

У користуванні даним комплексом є альтернатива використання додаткових фізичних пристроїв. Задачі ідентифікації, що покладені на вищеописані пристрої може виконувати і телефон через веб-застосунок. Для цього необхідно мати камеру у смартфоні. Вона може розпізнати штрих-код, тобто підтримує базовий метод ідентифікації товарів. Далі, через інтерфейс веб-застосунку можна буде зробити всі необхідні дії з товаром – додати його після покупки з магазину або видалити перед викиданням у кошик для сміття.

Отже, в залежності від потреб користувача існує три варіанти комплектації системи:

* Система без апаратної складової.
* Система з апаратною ідентифікацію штрих-кодів
* Система з використанням радіочастотної ідентифікації.

## 4.2 Алгоритм взаємодії користувача з комплексом

Розглянемо алгоритм взаємодії користувача з кожним варіантом комлектації системи.

### 4.2.1 Система без апаратної складової.

Система без апаратної складової буде мати обмежену функціональність, оскільки ідентифікація товарів буде здійснюватися користувачем вручну через камеру смартфону. Користувачу доступна взаємодія з системою тільки через веб-застосунок. Для того щоб додати товар до підсистеми моніторингу, користувачу необхідно перейти до сторінки додавання товару та камерою смартфону розпізнати штрих-код на упаковці, після чого товар буде додано до системи. Для того, щоб позначити товар як використаний, користувачу потрібно перейти в веб-застосунку по навігації до сторінки видалення товарів та сканувати ще раз камерою смартфону штрих-код на упаковці.

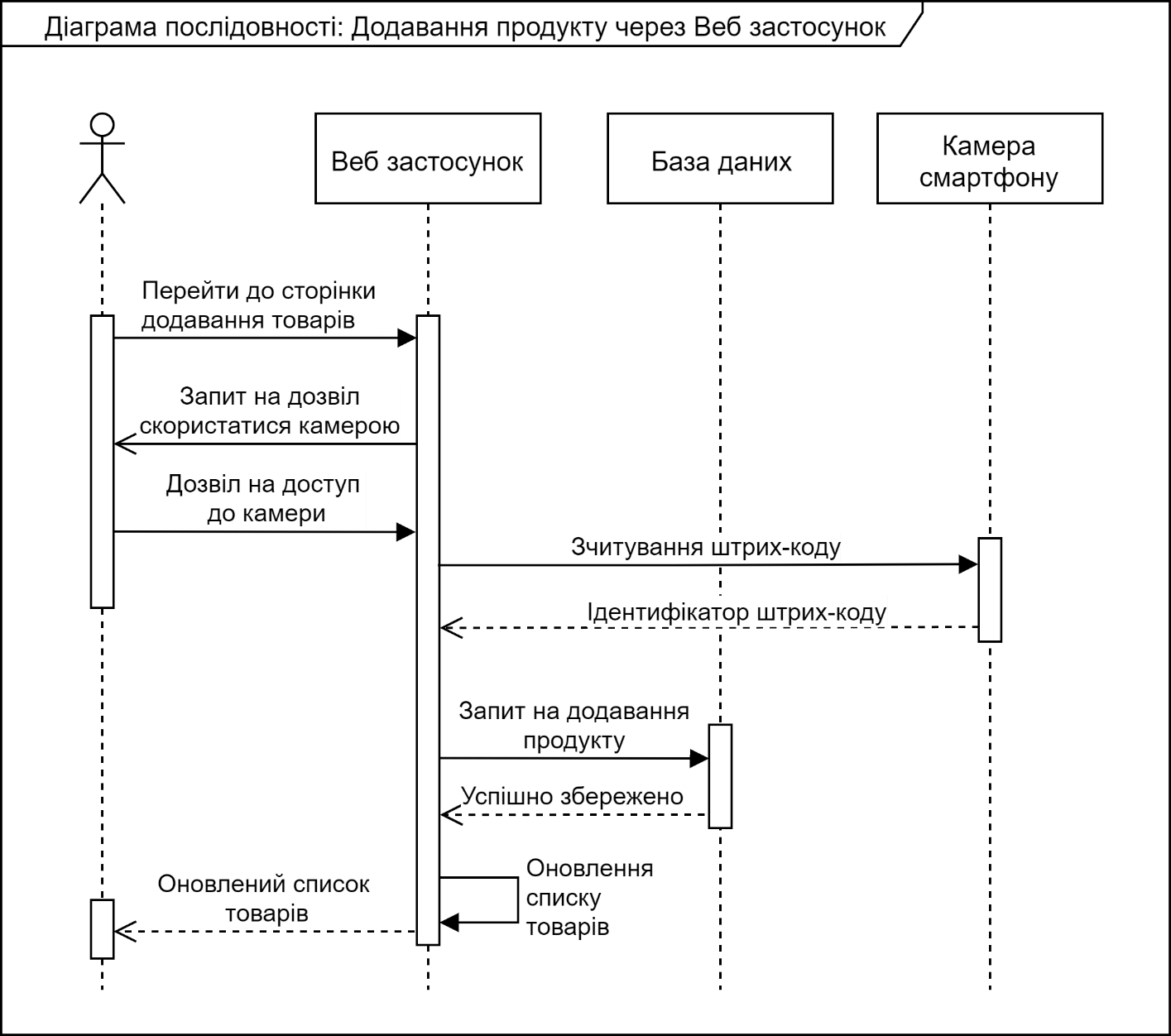


Рисунок 1. UML діаграма послідовності: Додавання продукту через веб-застосунок.

### 4.2.2 Система з апаратною ідентифікацію штрих-кодів.

Система з апаратною ідентифікацію штрих-кодів має в собі ті самі можливості, що і в попередньому пункті, але процес видалення товару буде автоматизований використанням апаратної підсистеми ідентифікації товарів за штрих-кодом. Апаратна підсистема представляє собою невеликий пристрій зі сканером штрих-коду, що потрібно розмістити поруч із кошиком для сміття. Перед тим як використану упаковку з-під товару викинути до кошика, користувачу необхідно сканувати штрих-код на упаковці. Така дія вже не вимагає використання смартфону для видалення товару з підсистеми моніторингу. Також слід зазначити, що розпізнавання штрих-коду спеціалізованим сканером здійснюється в рази швидше ніж камерою смартфону. Така комплектація та організація системи вже створює необхідні умови для того, щоб людина витрачала менше часу на видалення товару в підсистемі моніторингу.

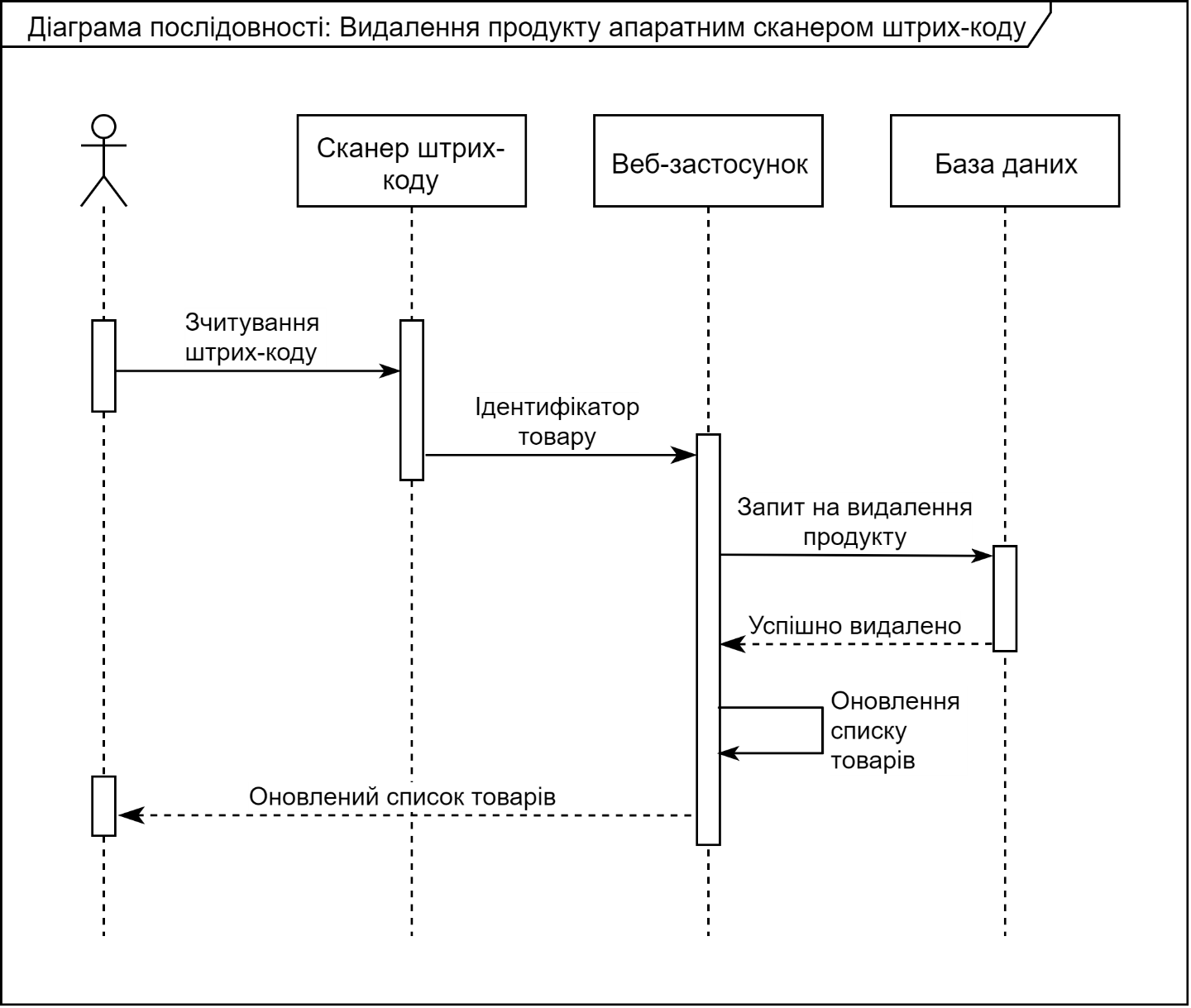


Рисунок 1. UML діаграма послідовності: видалення продукту апаратним сканерои штрих-коду.

### 4.2.3 Система з використанням радіочастотної ідентифікації

Система з використанням радіочастотної ідентифікації включає в себе всі можливості, що і описаний попередньо варіант. Але в пристрій, що був побудований на базі сканеру штрих-кодів, додається сканер RFID міток. Розміщення такої апаратної підсистеми видалення залишається незмінним – пристрій знаходиться біля кошику для сміття. Тільки тепер він може спрацьовувати як на штрих-код, так і на RFID мітку. Якщо в радусі дії RFID рідера з’являється мітка, то він миттєво її зчитує і відправляє дані ідентифікатора на сервер для подальшої обробки.

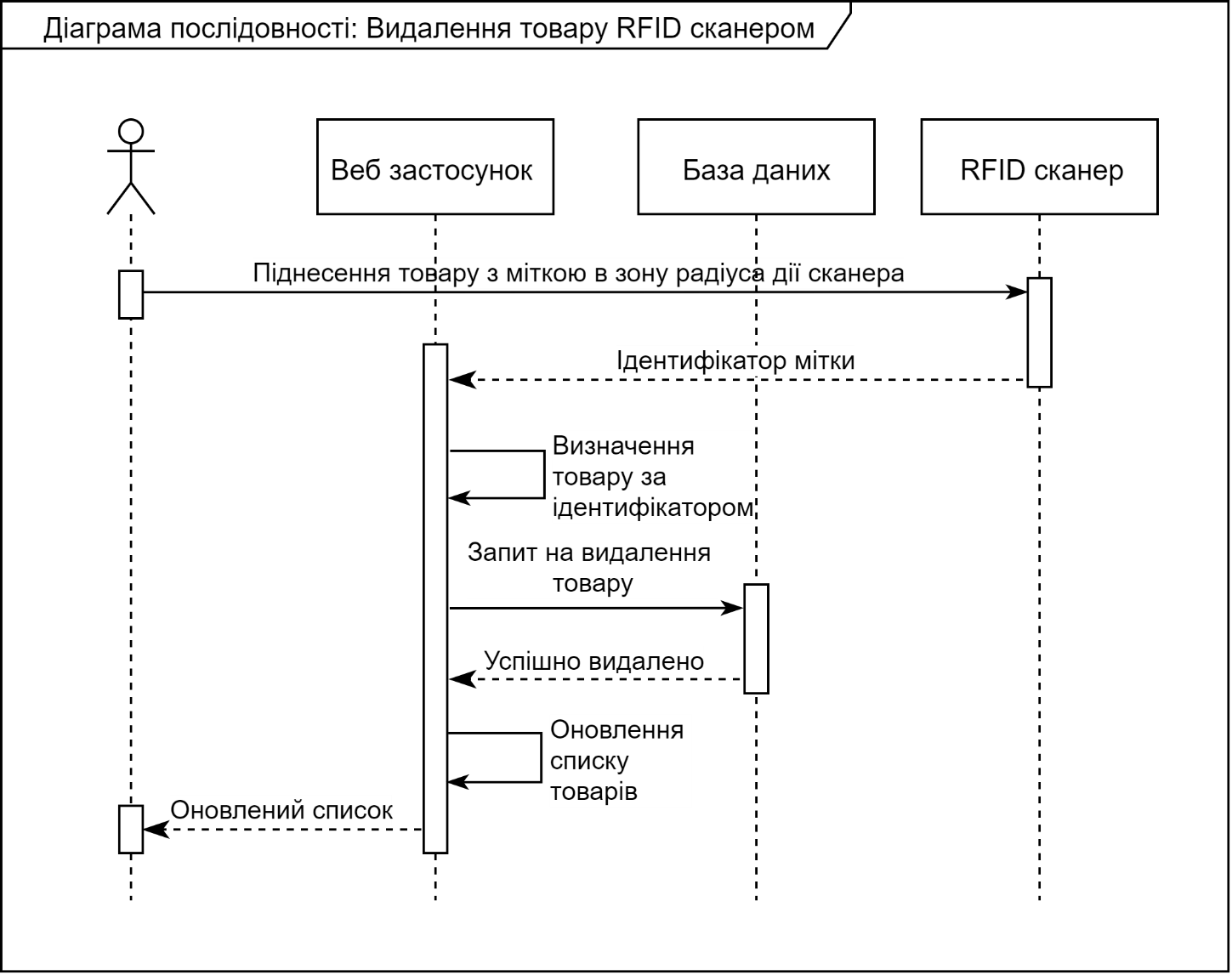


Рисунок 1 – UML діаграма послідовності: видалення товару RFID сканером

Проблему, відсутності маркування радіочастотними мітками упаковки товарів вирішує ще один пристрій. Це також зчитувач RFID міток. Задача цього пристрою полягає в тому, щоб зареєструвати радіочастотну мітку в системі. Для цього потрібні мітки-наліпки. Кожна відсканована цим пристроєм мітка стає відомою підсистемі моніторингу, але вона на є прив'язана до жодного товару. Назвемо таку мітку вільною. Для того щоб прив'язати мітку до товару, користувач наліплює зіскановану мітку на упаковку цього товару. Наступною дією у веб-застосунку, навпроти відповідного товару, він натискає відповідну кнопку що зв'язує обраний товар з вільною міткою. Тепер системі відомо що мітка з відповідним ідентифікатором прив'язана до обраного товару.

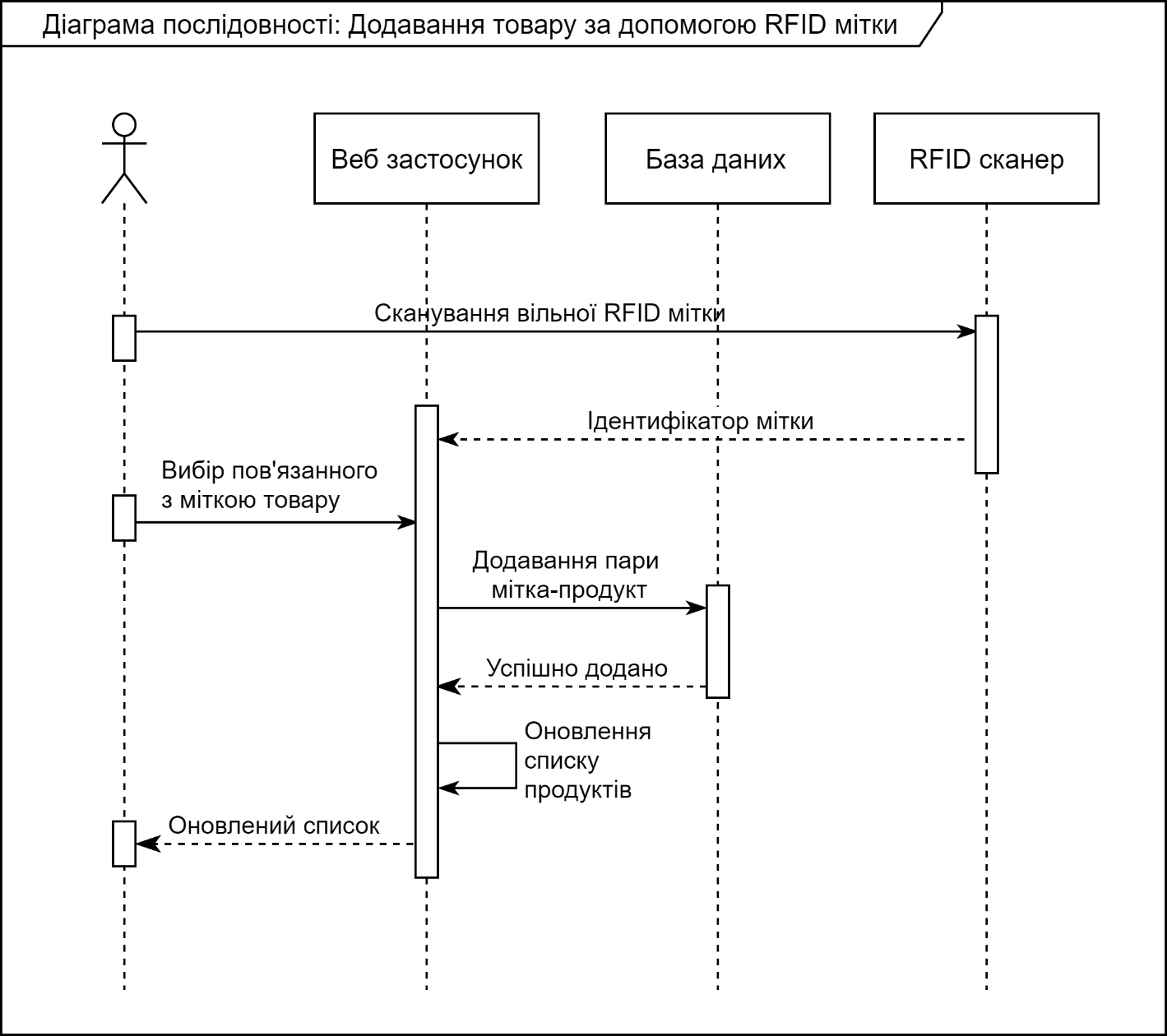


Рисунок - UML діаграма послідовності: Додавання товару за допомогою RFID мітки

Таким чином користувач має маркувати всі товари, що в нього є в наявності для змоги користуватися перевагами радіочастотної ідентифікації. Тепер, під час викидання упаковки у сміття, користувач не повинен робити жодних дій для зчитування мітки RFID сканером. Пролітаючи у повітрі повз сканер, мітка зловить радіохвилі, що ним випромінюються, і у відповідь відправить свій ідентифікатор. Сканер, отримавши ідентифікатор, передає дані до серверу через Інтернет до програмної частини системи. Оскільки в системі зберігається інформація про всі зв’язані ідентифікатори міток і товарів, відповідний товар буде вважатися утилізованим. Ця інформація відобразиться на сторінці моніторингу товарів. Якщо RFID мітки закінчилися, система може продовжити працювати за сценарієм попередньо-описаного варіанту комплектації – з ідентифікацією лише за штрих-кодами. Слід також зазначити, що одна і та сама мітка може бути використана ще раз. Як тільки товар було відскановано та викинуто у сміттєвий кошик, мітка була видалена з системи моніторингу. Тобто цю мітку можна буде відчепити від упаковки, та зв’язати із новим товаром, повторивши ще раз алгоритм додавання за допомогою RFID мітки, зображений на рисунку [?].

## 4.3 Реалізація програмної частини

Розглянемо більш детально базову модель використання розроблюваного комплексу. В усіх наведених вище сценаріях використання необхідно взаємодіяти з веб-застосуноком. Це є невід'ємна частина розроблюваної системи. Веб-застосунок поєднує в собі, як ядро системи, так і клієнтський застосунок.

Програмна частина повинна забезпечувати обчислення та аналіз отриманої інформації, синхронізацію даних між пристроями користувачів та бути базовим інструментом автоматизації ведення домашнього господарства.

Завдяки розвиненим API сучасних браузерів, веб-застосунки можуть взаємодіяти майже з усіма апаратними складовими смартфонів та ноутбуків. Для базової реалізації ідентифікації продуктів за допомогою штрих-кодів було вирішено максимально використовувати ті інструменти, що вже є наявні у більшості потенційних користувачів. Так, наприклад у більшості користувачів є смартфони. В свою чергу у всіх смартфонів є камера. Тож для ідентифікації товару необхідно лише зробити захват зображення штрих-коду з камери смартфону та декодувати його програмними засобами.

З цього виходить, що смартфон є повноцінним та самодостатнім інструментом взаємодії користувача з системою автоматизації ведення домашнього господарства. Однак це накладає деякі обмеження. Так, наприклад, необхідно завжди тримати смартфон при собі. Щоб у будь-який момент часу можна було його дістати та зробити відповідні дії з товарами. Тобто, якщо користувач захоче викинути використану упаковку у кошик для сміття, йому потрібно буде спочатку достати смартфон, відсканувати штрих-код камерою смартфону та, лише після цього, викинути упаковку до сміттєвого кошику.

Для розробки програмного забезпечення була обрана платформа .NET Core. Ця платформа є крос-платформеним спадкоємцем платформи .NET Framework від Microsoft. Це означає, що розробку та хостинг застосунків можна буде здійснювати на всіх операційних системах для комп’ютерів: Windows, MacOS, Linux. Для розробників це означає, що вони не будуть скуті вибором машини для розробки програмного забезпечення. Для бізнесу це буде означати, що вони можуть розміщувати застосунки на хостингах та серверах з будь-якою операційною системою. Слід зауважити, що сервери під управлінням Windows в середньому виходять дорожче за аналогічні сервери з іншою операційною системою. Це було одним з ключових недоліків платформи .NET Framework коли мова йшла про вибір платформи для розробки програмного забезпечення.

В рамках платформи .NET Core існує чотири сценарія розробки застосунків:

* Веб застосунки ASP.NET Core
* Консольні застосунки
* Допоміжні бібліотеки та фреймворки
* Застосунки для Windows 10 (UWP – Universal Windows Platform)

Серед усіх інших інструментів створення веб-застосунку було обрано ASP.NET Core. Це потужний фреймворк в рамках платформи .NET Core що забезпечує розробника повним стеком технологій для розробки повноцінного веб-застосунку. Він включає в себе наступні компоненти для розробки:

* Entity Framework (EF) Core
* Identity Core
* MVC Core
* Razor Core
* SignalR
* Blazor

В розробці програмного застосунку було використані всі компоненти фреймворку ASP.NET Core окрім SignalR та Blazor. Короткий опис кожного з них наведено нижче.

Entity Framework (EF) Core – це інструмент роботи за базою даних, так званий object-relational mapper (ORM). Він представляє таблиці бази даних та записи в них, у вигляді звичних для розробника класів та об’єктів відповідно. Це значно полегшує взаємодію з базою даних.

Entity Framework Core може взаємодіяти з різними СУБД. Для цього платформа .NET Core надає велику кількість провайдерів для роботи з базами даних. Серед них є провайдери для роботи з наступними базами даних: MySQL, MariaDB, MS SQL Server, PostgreSQL, SQLite, Oracle DB та ін. Для того, щоб не залежати від операційної системи розробки та хостингу було вирішено використати MySQL, оскільки ця СУБД має широку підтримку та зручний інтерфейс для розробки. За допомогою засобів ASP.NET Core та EF Core було створено модель бази даних за принципом code-first: спочатку були створені класи для кожної моделі та сутності бази даних, а потім інструментами EF Core був сгенерований скрипт для створення відповідної схеми бази даних. Схему бази даних наведено на кресленнику []

Identity Core - це система авторизації та аутентифікації засобами графічного інтерфейсу у веб-застосунку. За її допомогою користувачі можуть створити обліковий запис, що буде зберігатися в базі даних. Також Identity Core може бути інтегрований із зовнішнім постачальником послуг аутентифікації. Наразі підтримуються наступні зовнішні постачальники послуг: Facebook, Google, Microsoft та Twitter. В таким спосіб користувачу не має потреби реєструвати обліковий запис на свою пошту та запам’ятовувати ще один пароль. Уся необхідна інформація для аутентифікації буде отримуватися з облікового запису вище перелічених постачальників послуги аутентифікації.

Identity Core створює свої таблиці в базі даних для того, щоб зберігати інформацію про ролі, права, токени для надання доступу та інформацію з облікових записів користувачів. В веб-застосунку створюються два контролери, що відповідають за всі дії пов’язані з роботою з обліковим записом: логін та реєстрація аккаунту, відновлення паролю, редагування особистої інформації, інтеграція із зовнішніми постачальниками послуг аутентифікації та навіть впровадження двох-факторної аутентифікації через мобільний пристрій. При реєстрації кожному користувачу присвоюється ідентифікатор в строковому форматі, що зберігається в таблиці aspnetusers. Саме за цим ідентифікатором в подальшому буде встановлюватися зв’язок між веб-застосунком, користувачем та тими товарами, що знаходяться в підсистемі моніторингу.

MVC Core та Razor Core – це фреймворки для побудови клієнтських щастосунків. Вони схожі, але є в них відмінність у концепції побудови архітектури. Виходячи з назви зрозуміло, що MVC Core надає інструмент побудови веб-застосунків та API використовуючи шаблон розробки Model-View-Controller (MVC). Це архітектурний шаблон, що розділяє застосунок на три основні групи компонентів: модель (model), представлення (view) та контролер (controller). Такий поділ допомагає досягти такого поняття у програмуванні, як розділення відповідальності (separation of concerns).

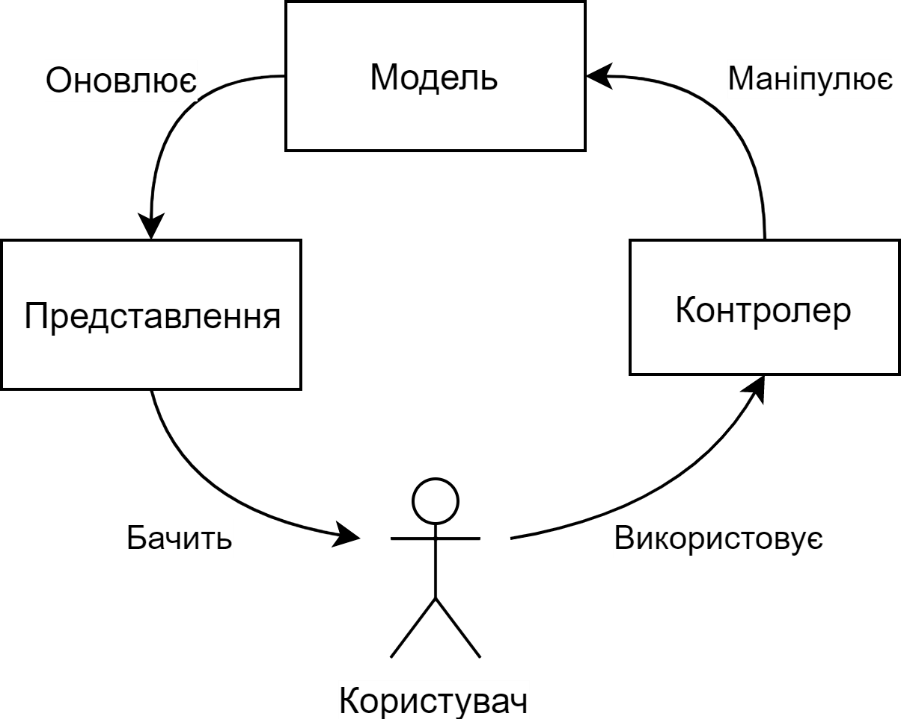


Рисунок – Взаємодія компонентів в шаблоні MVC

Razor Core -це технологія що з’явилась нещодавно в платформі .NET. Навідміну від MVC Core полягає в тому, що модель і контролера вже включені в саму сторінку застосунку. Інакше кажучи, шаблон приймає вигляд Model-View-ViewModel (MVVM). Така модель забезпечує двонаправлений рух даних у застосунку. Якщо дані були змінені у моделі, то оновиться представлення, і навпаки (у MVC тільки однонаправлений рух даних – від моделі до представлення).

### 4.3.1 Опис методів API

Програмна частина комплексу надає публічну API для роботи з віддаленими пристроями та слугує веб сервером для клієнтських застосунків. API (application programming interface) – це інтерфейс або протокол зв'язку між клієнтом і сервером, призначений для спрощення побудови програмного забезпечення на стороні клієнтського застосунку. Цей інтерфейс має заздалегідь визначені методи та физначенй формат відповіді на запити до цих методів. Таким чином, при розробці програмного забезпечення для апаратних пристроїв, розробник знає, що та в якому форматі йому треба відправити до сервера щоб отримати бажану відповідь.

API розроблюваної системи побудована за допомогою MVC Core та представляє собою виділенний контролер, що надає наступні публічні методи для взаємодії з апаратною частиною та клієнтським застосунком:

* GetProduct
* AddUserProduct
* AddUnassignedRfid
* BindUnassignedRfidToUserProduct
* UserProductToBin
* UserProductToBinByBarcode
* UserProductToBinByRfid

Нижче наведено опис кожного методу API та всі необхідні параметри для роботи з ним.

#### 4.3.1.1 GetProduct

GetProduct - Надає дані про товар: його ідентифікатор, назву та короткий опис.

HTTP запит: GET /api/get-product

Параметри URL запиту

|  |  |
| --- | --- |
| code | Тип: string |
| Опис: штрих-код товару |

Якщо запит виконано успішно, тіло відповіді буде мати наступний вигляд у JSON форматі:

{

success: <bool>,

product: {

id: < ідентифікатор продукту > ,

title: < назва продукту > ,

description: < опис продукту >

}

}

#### 4.3.1.2 AddUserProduct

AddUserProduct – Додає товар до підсистеми моніторингу для відповідного користувача.

HTTP запит: POST /api/add-user-product

Параметри URL запиту

|  |  |
| --- | --- |
| userId | Тип: string |
| Опис: ідентифікатор користувача |
| productId | Тип: int |
| Опис: ідентифікатор товару |
| quantity | Тип: int |
| Опис: кількість одиниць товару |

Якщо запит виконано успішно, тіло відповіді буде мати наступний вигляд у JSON форматі:

{ success: true }

#### 4.3.1.3 AddUnassignedRfid

AddUnassignedRfid – Додає вільну RFID мітку до системи. В такому стані мітка очікує, поки користувач не прив’яже її до конкретного товару через інтерфейс веб-застосунку.

HTTP запит: POST /api/add-unassigned-rfid.

Параметри URL запиту:

|  |  |
| --- | --- |
| rfid | Тип: string |
| Опис: ідентифікатор RFID мітки |
| userId | Тип: string |
| Опис: ідентифікатор користувача |

Якщо запит виконано успішно, тіло відповіді буде мати наступний вигляд у JSON форматі:

{ success: true }

#### 4.3.1.4 BindUnassignedRfidToUserProduct

BindUnassignedRfidToUserProduct – Зв’язує вільну мітку з наявним товаром у підсистемі моніторингу для поточного користувача. Відповідний запис додається до бази даних

HTTP запит: POST /api/bind-unassigned-rfid-to-user-product.

Параметри URL запиту:

|  |  |
| --- | --- |
| userProductId | Тип: int |
| Опис: ідентифікатор запису у підсистемі моніторингу, до буде прив'язана RFID мітка |

Якщо запит виконано успішно, тіло відповіді буде мати наступний вигляд у JSON форматі:

{ success: true, message: "RFID мітка усішно прив’язана до товару" }

#### 4.3.1.5 UserProductToBin

UserProductToBin – Видаляє з системи моніторингу певну кількість товарів, що пов’язані з певним користувачем. Якщо більше немає таких товарів (усі одиниці товару було викинуто до сміттєвого кошика), запис повністю видаляється з бази даних.

HTTP запит: POST /api/user-product-to-bin.

Параметри URL запиту:

|  |  |
| --- | --- |
| userId | Тип: string |
| Опис: ідентифікатор користувача |
| productId | Тип: int |
| Опис: ідентифікатор товару |
| quantity | Тип: int |
| Опис: кількість одиниць товару |

Якщо запит виконано успішно, тіло відповіді буде мати наступний вигляд у JSON форматі: { success: true }

#### 4.3.1.6 UserProductToBinByBarcode

UserProductToBinByBarcode виконує ті самі дії, що і метод UserProductToBin за тим виключенням, що критерієм пошуку запису в базі даних є штрих-код.

HTTP запит: POST /api/user-product-to-bin-by-barcode.

Параметри URL запиту:

|  |  |
| --- | --- |
| userId | Тип: string |
| Опис: ідентифікатор користувача |
| barcode | Тип: string |
| Опис: штрих-код товару |

Якщо запит виконано успішно, тіло відповіді буде мати наступний вигляд у JSON форматі:

{ success: true }

#### 4.3.1.7 UserProductToBinByRfid

UserProductToBinByRfid – Виконує ті самі дії, що і метод UserProductToBin. Критерієм пошуку запису для видалення є інший запис, що пов’язує RFID мітку та товар. Цей запис знаходиться за переданим до запросу ідентифікатором мітки. Після успішного видалення товару з моніторингу для поточного користувача, запис, що пов’язує RFID мітку та товар, видаляється з бази даних. RFID мітка з відповідним ідентифікатором може бути знову використана для прив’язання до іншого товару.

HTTP запит: POST /api/user-product-to-bin-by-rfid.

Параметри URL запиту:

|  |  |
| --- | --- |
| rfid | Тип: string |
| Опис: ідентифікатор RFID мітки |

Якщо запит виконано успішно, тіло відповіді буде мати наступний вигляд у JSON форматі:

{ success: true }

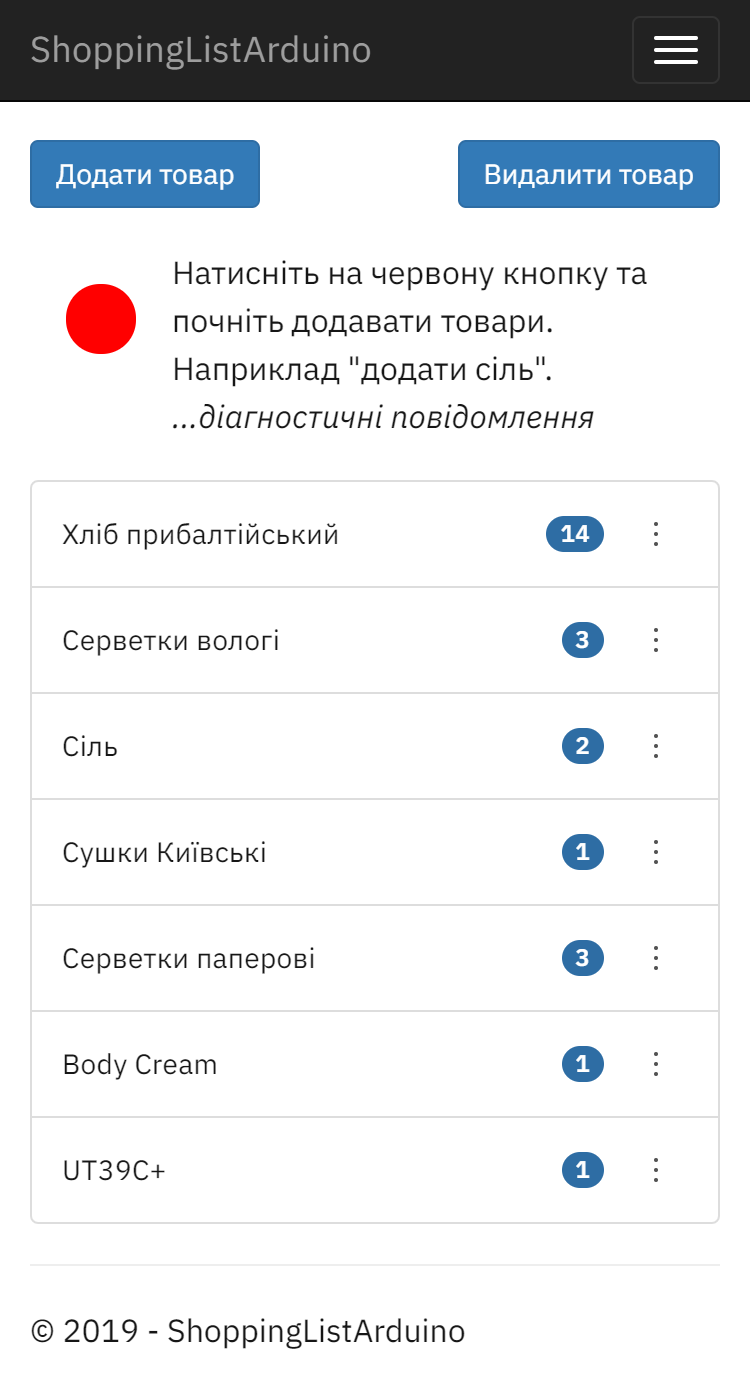
За допомогою вищеописаних методів реалізована вся бізнес-логіка програмно-апаратного комплексу автоматизації. Всі пристрої, що так або інакше взаємодіють з товарами (камера смартфону, сканер штрих-коду, сканер RFID), надсилають отримані дані в якості параметрів до методів API через Інтернет.

### 4.3.2 Клієнтський веб-застосунок

Клієнтська частина представлена у вигляді веб-застосунку, що складається з кількох сторінок.

Головна сторінка є основною та містить у собі посилання на сторінки для додавання та видалення товарів через камеру смартфону, список наявних в домі товарів на поточний момент часу та кнопку для запису голосових команд. Список містить у собі записи з бази даних для товарів, що є в підсистемі моніторингу для поточного користувача. Кожен елемент списку складається найменування товару та кількості наявних одиниць цього товару. В кінці кожного рядка є функціональне меню, що складається з наступних елементів:

* Прив’язати RFID мітку. При натисканні на цей пункт меню, якщо в системі існує вільна RFID мітка, то вона буде зв’язана з обраним товаром. Відповідний запис буде додано до бази даних.
* Редагувати. Ця дія перенаправляє користувача на строрінку редагування, де можна змінити кількість наявних товарів. Це найпростіший засіб додавання або видалення якогось товару в процесі ведення домашнього господарства.
* Видалити. Ця дія повністю видаляє запис про обраний товар з бази даних для поточного користувача.

’

Веб-застосунок підтримує управління голосом. Таке управління може виявитись для користувачів більш зручним, ніж послідовність дій, пов'язаних із скануванням штрих-коду за допомогою камери смартфону. Особливо навявність такого функціоналу полегшить користування мобільним веб-додатком людям з обмеженими можливостями. Для забезпечення базової взаємодії з товарами, керування відтримує дві команди: на додавання та на видалення товару з підсистеми моніторингу. Для коректного виконання команди необхідно чітко вимовити найменування товару саме так, як воно відображається в веб-застосунку. Томущо пошук товару буде здійснюватися саме за назвою, а не за ідентифікатором що нанесений на упаковці товару.

Щоб здійснити голосове управління, необхідно на головній сторінці натиснути червону кнопку. Це призведе до активації мікрофону пристрою та початку запису голосвої команди. Наступним кроком людина має вимовити команду згідно шаблону, що складається з дії (додавання або видалення) та найменування товару, до якого необхідно застосувати цю дію. У спливаючому вікні буде відображено результат розпізнавання команди. Оскільки навіть сучасні алгоритми не гарантують точного розпізнавання людської мови українською, такий спосіб підтвердження дії є доречним. Якщо команду було розібрано правильно – до API веб-застосунку автоматично будуть здійснені відповідні запити. Користувачу лише буде відображена інформація про успішне виконання операції. Якщо голосова команда була розпізнана, але не відповідала вищеописаному шаблону команди, користувача буде сповіщено про це, та відображено на головній сторінці текст, що було розпізнано.

Для реалізації розпізнавання людської мови, в веб-застосунку було використано Web Speech API. Web Speech API складається з двох частин: Speech Synthesis та Speech Recognition.

Speech Synthesis доступна розробнику через інтерфейс SpeechSynthesis. Це, так званий, text-to-speech компонент, що дозволяє програмам читати людською мовою текст. Зазвичай це відбувається через вбудовані динаміки. Аудіозапис може бути створений з використанням різних типів голосів та з емоційним окрасом. Наприклад, напівжирний текст може бути виділено інтонацією.

Speech Recognition доступна розробникам через інтерфейс SpeechRecognition, що надає можливість розпізнати голосовий контекст з аудіозапису, зробленого мікрофоном пристрою. Конструктором інтерфейсу створюється об'єкт SpeechRecognition, що виявляє, в аудіозаписі з’являється людьска мова. Інтерфейс SpeechGrammar являє собою контейнер для певного набору граматик, який повинен розпізнавати веб-застосунок. Граматика визначається за допомогою JSpeech Grammar Format (JSGF).

JSpeech Grammar Format – це кросплатформенний спосіб представлення граматик для роспізнавання людської мови. Ці граматики використовються системами розпізнавання речі для того, щоб визначати що саме необхідно слухати. Інакше кажучи, за домопомогою граматики створюється шаблон повідомлення, по якому орієнтується система розпізнавання людської мови.

Слід зазначити, що для браузерів Chrome, використання інтерфейсу SpeechRecognition використовує серверні ресурси для розпізновання. Тобто аудіо буде відправлено до веб сервісу для обробки та розпізнавання. Тож ця операція розпізнавання потребує доступ до Інтернету.

Для ідентифікації товару за штрих-кодом через веб-застосунок, необхідно перейти до відповідної сторінки. Додавання товару до системи здійснюється тільки через камеру смартфону. Для цього користувач має перейти за посиланням «Додати товар». Якщо запит на сторінку був зробленій за захищеним протоколом, то користувач отримає запит на дозвіл скористатися камерою пристрою для захвату зображення. Після надання користувачем доступу, веб-застосунок почне отримувати інформації з камери смартфону. Коли в поле зору камери потрапить штрих-код, він буде розпізнаний.

Оскільки програмні засоби розпізнають штрих-код значно гірше за апаратні, користувач побачить спливаюче вікно. У ньому буде міститися інформація про розпізнаний ідентифікатор. Для продовження процесу користувач має підтвердити, що код було розпізнано без помилок. В зворотньому випадку застосунок спробує розпізнати його знов. Слід зазначити, що на якість розпізнавання сильно впливає якість камери смартфону. Тож етап перевірки правильності розпізнання має бути невід'ємною частиною процесу додавання товару засобами смартфону. Описаний вище етап взаємодії користувача з застосунком зображено на рисунку [].

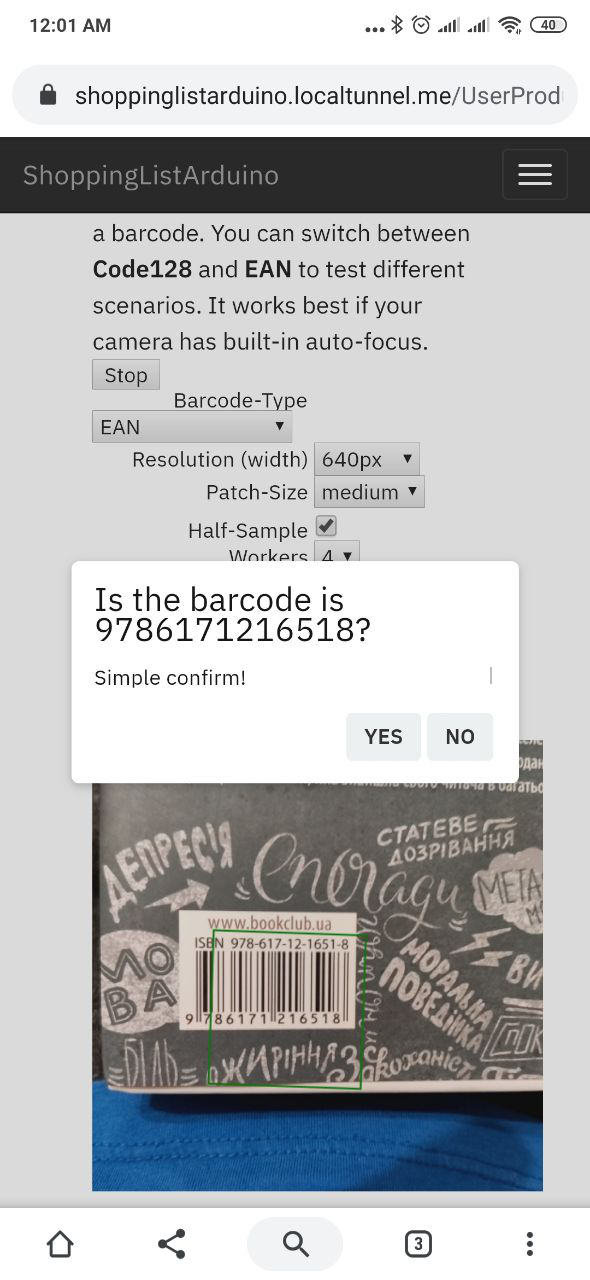


Рисунок – Зчитування штрихкоду кмерою смартфону через веб-застосунок

На сторінці, де існує взаємодія з камерою можна вибирати параметри зображеня, що надає пристрій: розподільна здатність зйомки, включений або виключений діод для освітлення, фронтальний чи основний модуль камери. Розпізнається зображення за допомогою використання Quagga JS.

QuaggaJS – це javascript бібліотека, що виконує функції сканеру штрих-коду. Вона повністю написана на JavaScript, та підтримує декодування багатьох типів штрих-кодів, таких як: EAN, CODE 128, CODE 39, EAN 8, UPC-A, UPC-C, I2of5, 2of5, CODE 93 та CODBAR. Для розпізнавання у метод бібліотеки потрібно передати зображення штрих-коду. Відмінністю цієї бібліотеки від інших є можливість використовувати getUserMedia API, що надається сучасними браузерами, для отримання прямого доступу до камери пристрою.

Важливим є той факт, що getUserMedia API потребує забезпечення захищеного доступу до веб-застосунку через браузер [<https://groups.google.com/forum/#!topic/discuss-webrtc/sq5CVmY69sc>]. Таке обмеження дуже доречно в контексті того, що через Інтернет буде передаватись зображення з камери користувача. Тож вкрай важливим є забезпечення шифрування даних що будуть передаватися по мережі. Адже в зворотньому випадку це компроментує місцезнаходження користувача та іншу особисту інформацію, що може потрапити до кадру.

Навіть за умовик захищеного з’єднання getUserMedia API не буде активовано, поки користувач не дозволить веб-застосунку використати камеру свого пристрою. Кожен браузер забов’язаний показати індикатор про те, що камера або мікрофон знаходяться у використанні конкретної веб-сторінки. Такий ідентифікатор працює окремо від апаратних індикаторів, що може мати пристрій. Наприклад, як світлодіо поруч з об’єктивом фронтальної камери на ноутбуках. В браузерах Firefox, в рядку URL-адрес відображається пульсуюча червона піктограма. Вона вказує на те, що запис ведеться. Піктограма стає сірого кольору, якщо дозвіл є, але запис наразі не ведеться. Якщо ви відключили камеру, індикатор активності камери згасне, вказуючи на те, що камера не здійснює запис відео. При цьому, дозвіл на використання камери не відміняється. Тобто запис продовжиться, коли камера буде підключена знову.

Аналогічну будову до розглянутої сторінки додавання товарів має сторінка для видалення товарів з підсистеми моніторинга через камеру смартфону. Посилання на цю сторінку також знаходиться на головній сторінці. Структура та реалізація майже така сама як і у розглянутій попередньо сторінці додавання товару. Тільки після успішного сканування штрих-коду, за отриманим ідентифікатором буде знайдено відповідний запис у базі даних та змінено.

Недолік штрих-коду полягає в тому що він містить в собі тільки ідентифікатор товару. Так було зроблено, оскільки його фізичний розмір має пряму залежність від об'єму інформації, що в ньому міститься. З цього виходить, що сторона, що сканує штрих-код має самостійно визначити за отриманим ідентиіфкатором, якому товару належить отриманий ідентифікатор. Це означає, що має існувати база даних, де міститься запис про товари з їх штрих-кодами. Окрім ідентифікатору та найменування товару, в базах даних повина міститися інша внутрішня інформація, що стосується організації або підприжємства на якому було скановано цей товар. Наприклад, супермаркети маєть відповідні баз даних, де, окрім ідентифікаторів і найменувань, вірогідно зберігаються дані про кількість та роздрібну ціну товарів. Ця інформація є комерційною тайною магазину, і не може бути викладена у вільний доступ. Аналогічно для багатьох інших компаній, що мають зв’язок з ідентифікацією товарів за штрих-кодами, інформація в базах даних є прихованою.

Існують закордонні відкриті бази даних штрих-кодів, але в них не міститься інформації про розповсюджені в Україні товари. Наприклад, одна з найвідоміших баз штрих-кодів International Barcodes Database, містить у собі лише 7 записів про товари, вироблені в Україні [<https://barcodesdatabase.org/selected-products/?country_list_select=UA&sort_options=0>]. Додавання товару до такої бази даних зводиться до певного процесу. Наприклад, щоб додати товар до бази Barecode Lookup, необхідно заповнити форму з 6 обов’язкових полів (всього в форма складається з 22 полів). І таку форму необхідно заповнити інформацією про кожен товару, що присутній в роздрібній торгівлі. Оскільки форма має заповнюватися вручну людиною-працівником того чи іншого виробника, це не є витратою часу та грошей для даної організації. Поки такі відкриті бази даних не будуть мати переваг перед внутрішніми приватними базами даних підприємств, що безпосередньо приймають участь у маркетинговому ланцюгу, розповсюдженням інформації про свої товари українські організації займатися не будуть. Треба відмітити, що для додавання своїх товарів до деяких відкритих баз даних, необхідно навіть заплатити гроші. Наприклад, для того, щоб додати товар до вищезгаданої International Barcodes Database, потрібно внести оплату в розмірі від 10$ за одиницю, якщо додається одразу більше 50 товарів та 20$ за одиницю – якщо додається тільки один товар [<https://internationalbarcodes.com/barcode-registration/>]. Також, тільки законим власник зареєстрованого штрих-коду у внутрішньому органі країни, що займається цим питанням. Для перевірки цього факту при реєстрації товару у базі, необхідно буде прикріпти відповідний документ [<https://barcodesdatabase.org/barcode-registration/>].

В першу чергу, розроблювана система орієнтована на використання у внутрішньому ринку України. Тож більшість штрих-кодів, що будуть скануватися, належатимуть українським виробникам. Для забезпечення веб-застосунку інформацією про найменування товарів та їх відповідному штрих-коду було подано відповідні запити в кілька мереж роздрібної торгівлі. На всі з них було отримано відмову.

Кожен день на внутрішньому ринку роздрібної торгівлі України можна зустріти все більше і більше нових товарів. Одні з них це результат плідної роботи українських підприємців, що починають виробляти нові продукти, інші – нові товари привезені з інших країн. Тож централізоване підтримання бази даних щтрих-кодів в актуальному стані є проблемою. Тож було прийняте рішення надати змогу всім користувачам веб-застосунку навповнювати власну базу даних інформацією про найменування товару, його корткий опис та ідентифікатор штрих-коду, наліплений на упаковці. Оскільки база даних єдина в межах всієї програмної частини комплексу, оновлені записи в таблиці товарів будуть доступні для всіх користувачів миттєво.

Для того, щоб додати новий товар до внутрішньої бази даних веб-застосунку, користувачу необхідно перейти на відповідну сторінку. Посилання на неї знаходиться в головному меню. На цій сторінці знаходиться список всіх найменувань в базі даних. Скріншот цієї сторінки зображено на рисунку [].

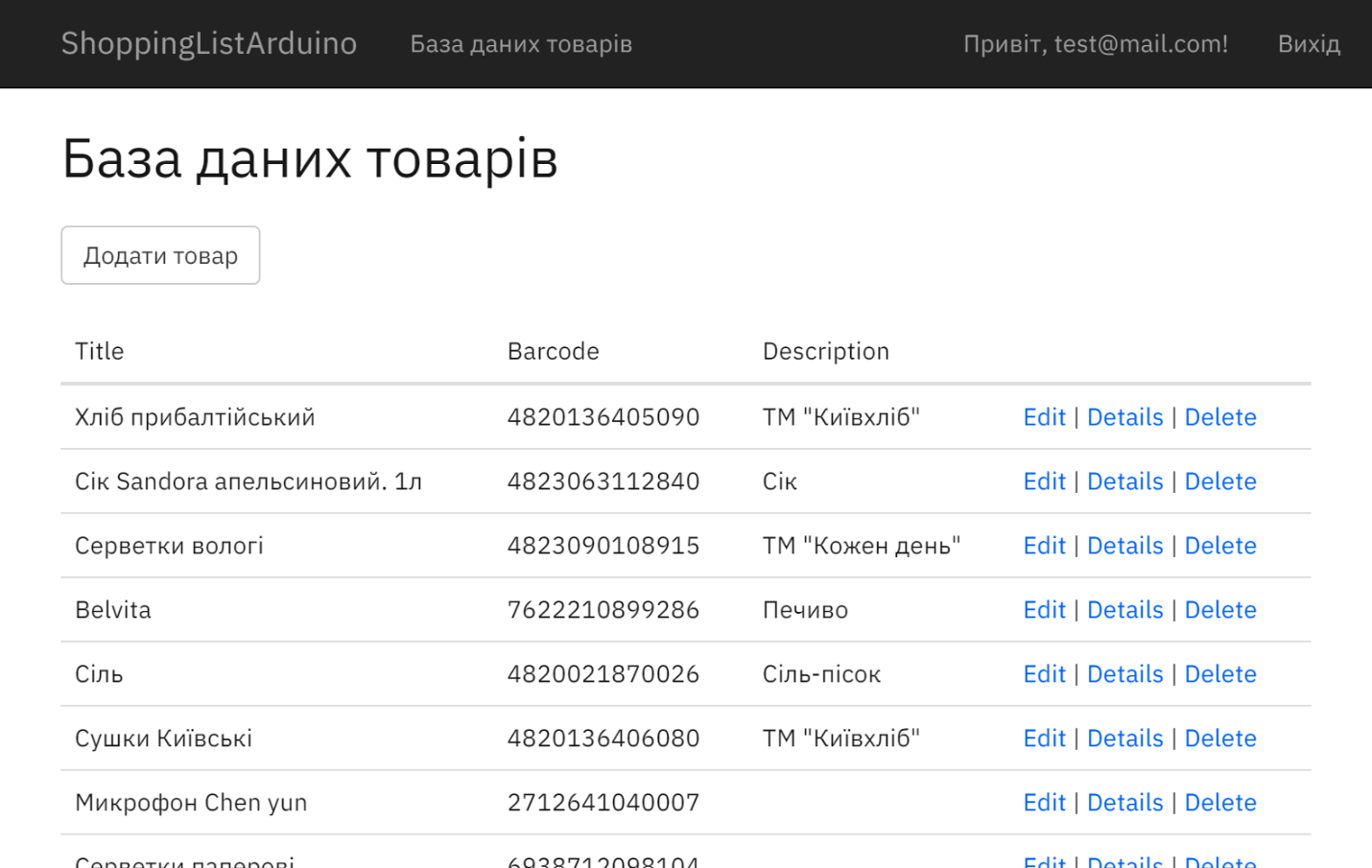


Рисунок – Сторінка існуючих товарів у бази даних веб-застосунку

Окрім списку товарів, що вже додані до бази даних, на сторінці розміщене посилання на сторінку для додавання нового товару. Перейшовши за цим посиланням, користувач має заповнити форму з 3 полів, що зображена на рисунку []. В цій формі необхідно вказати штрих-код, найменування та опис товару відповідно до кожного поля. Після того, як користувач відправить форму, доданий товар вже з’явиться в оновленому списку існуючих товарів в базі даних веб-застосунку.

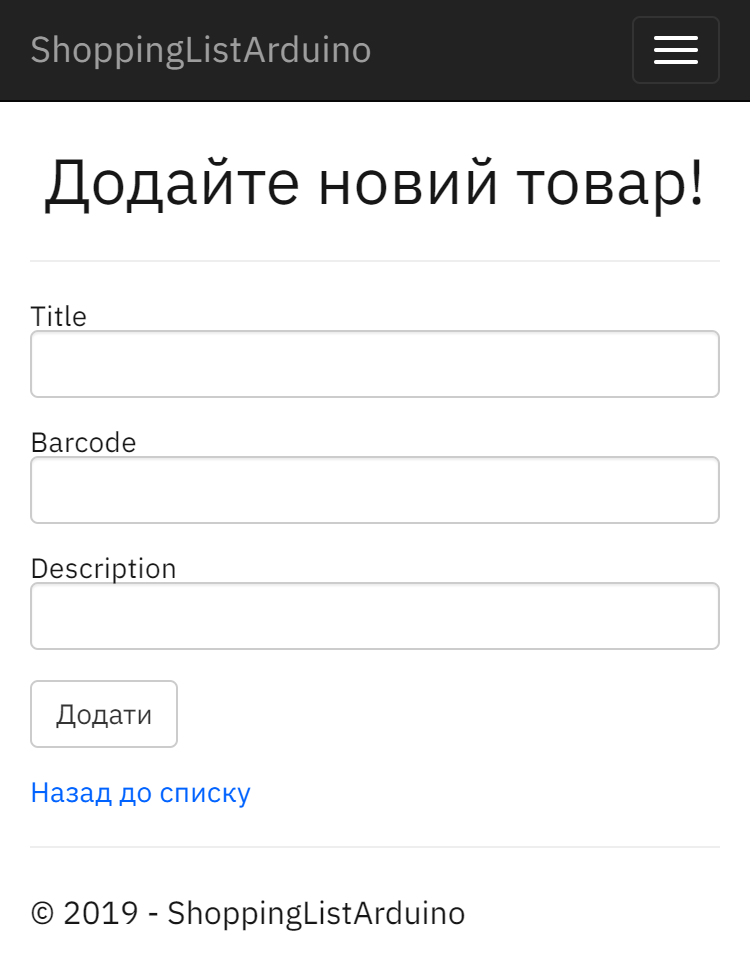


Рисунок – Форма додавання нового товару до бази даних веб-застосунку

## 4.4 Апаратна частина комплексу

Апаратна частина складається з трьох різних пристроїв. Вони створені для того щоб автоматизувати процес видалення товарів, після того як користувач викинув їх до сміттєвого кошика. Я було розглянуто вище, всі ці дії можна зробити і за допомогою одного лише смартфону через веб-застосунок. Але це створює додаткові незручності, оскільки в такому разі необхідною є умова наявності смартфону під рукою в момент кожної дії користувача. Всі пристрої апаратної складової поділяються на ті, що використовують ідентифікацію за штрих-кодом та ті, що використовують радіочастотну ідентифікацію. Розглянемо кожен пристрій окремо.

### 4.4.1 Пристрій ідентифікації за штрих-кодом

Пристрій ідентифікації за штрих-кодом має розміщуватися біля кошика для сміття. Його задача полягає в тому, щоб ідентифікувати товар, що було викинуто користувачем, за штрих-кодом на упаковці. Отриманий ідентифікатор має бути переданий як параметр до відповідного методу API веб-застосунку для здійснення подальших дій на програмному рівні системи.

Для того, щоб пристрій міг здійснювати ідентифікацію за штрих-кодом, його необхідно було оснастити відповідним сканером. Для базової ідентифікації буде достатньо можливостей світлодіодного сканеру штрих-кодів. В більшості випадків, такі сканери вже існують як самостійні пристрої, що вже готові до використання з інформаційними системами. Вони представляють собою ті пристрої, що можна зутсріти в будь-якому продуктовому. Зображення такого сканеру представлено на рисунку [].

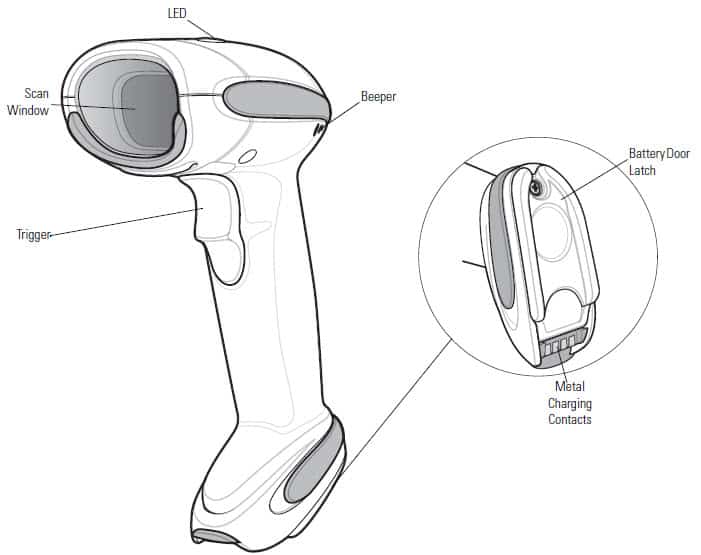


Рисунок – Світлодіодний сканер штрих-коду [https://www.carltontechnologies.com/wp-content/uploads/2016/09/Scanner-Parts.1.jpg]

Для задач, поставлених перед апаратним пристроєм у системі автоматизації ведення домашнього господарства таких сканер не підходить. Оскільки ві призначений для постійної взаємодії людини з ним. Він ручний, та має бути під'єднаний через безпровідним інтерфейсом до комп'ютеру або іншого обчислювального пристрою.

Однією з головних вимог до розроблюваного пристрою була компактність розмірів та стаціонарність, оскільки він має знаходитись постійно поруч з кошиком для сміття та не має бути об'єктом прямої взаємодії з користувачем. Для виконання поставленої задачі було знайдено окремий модуль сканеру штрих-кодів 1D Barcode Scanner E1005 виробництва Guangzhou YOKO Electron, зображаений на рисунку [].

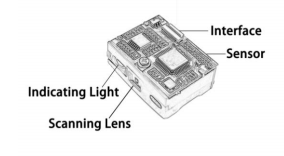


Рисунок – Світлодіодний сканер штрих-коду 1D Barcode Scanner E1005

E1005 має таку ж структуру, як і більшість сканерів. В середині корпусу знаходяться два світлодіоди, що є джерелами світла. Відбите від штрих-коду світло преломлюється через лінзу та потрапляє до фотосенсора, що за силою потоку світла генерує коливальний сигнал, оброблюється мікроконтроллером. На вихідний інтерфейс вже подається декодована інформація у вигляди байтової послідовності символів. Габарити модулю представлено на рисунку [].

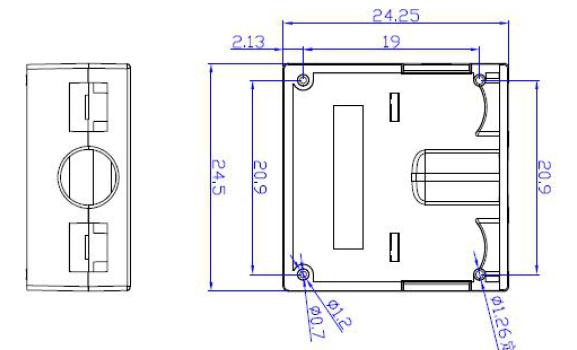


Рисунок – Габарити модуля сканеру E1005

Модуль сканера виконаний у невеликих розмірах. Його ширина та довжина не перевищує 25мм, що підходить для вищеописаних умов використання. Модуль оснащено 32-бітним чіпом власої розробки з запатентованим програмним забезпеченням компанії YOKO. Виробник стверджує, що сканер може декодувати штрих-коди з наявними невеликими завадами, такими як відблиски світла, пом’ята або кольорова поверхня упаковки товару.

Сканер підтримує інтерфейси комунікації з зовнішними пристроями через широко розповісюджені стандарти USB або RS-232. Комунікація через RS-232 здійснюється на швидкості від 1200біт/с до 115200біт/с. Стандартна конфігурація представляє собою щвидкість обміну даних 115200біт/с пакетами по 8 біт даних без біта парності та з одним стоп-бітом.

Зовнішній інтерфейс сканеру виконаний у вигляді конектору з 12 пінами EM1365-LD розробки компанії YOKO, що передає інформацію через станлартний роз’єм для гнучкого кабелю з 12 пінами та шагом 0.5мм. На таблиці [] наведено опис кожного піну модуля сканеру E1005.

|  |  |  |
| --- | --- | --- |
| № | Назва | Опис |
| 1 | NC | Резервний пін |
| 2 | VCC | Живлення постіним срумом та напругою 5В |
| 3 | GND | Заземлення |
| 4 | RX | Послідновний порт прийому |
| 5 | TX | Послідовний порт передачі |
| 6 | USB\_D- | USB сигнал |
| 7 | USB\_D+ | USB сигнал |
| 8 | NC | Резервний пін |
| 9 | BPR | Вихідний сигнал для дзвінка |
| 10 | LED | Вихід для індикатора успішного декодування |
| 11 | NC | Резервний пін |
| 12 | TRIG | Тригер для скануваня |

Таблиця – Таблиця пінів модуля сканеру E1005

Для активації сканеру – необхідно подати на 12 пін високий рівень сигналу. Для того, щоб сканер почав справцьовувати тільки тоді, коли користувач ввикидає сміття – було вирішено використати цифровий інфрачервоний дачтик виявлення перешкод YL-63, зображення на рисунку []. Його використовують, коли необхідно визначити наявність об’єкту, без точної інформації про дистанцію, на якій він знаходиться.



Рисунок – Датчик виявлення перешкод YL-63 [https://usamodelkina.ru/uploads/posts/2016-07/1469754633\_8199930e-d4e8-45dc-a925-2583d90021cc.jpg]

Датчик складається з інфрачервоного випромінювача та фотоприймача. Інфрачервоні хвилі відбиваються від перешкоди та фіксуються фотоприймачем. Цей пристрій побудовани на базі компаратора LM393, що видає високий сигнал на виході, якщо датчик фіксує перешкоду. Порогове значення визначається за допомогою потенціометра.

Датчик відноситься до класу дифузійних. В основі таких датчиків лежить фотоприймач, що визначає ступінь відбиття розсіяного світла. Це означає, що при виявленні перешкоди існує певна похибка. Вона залежить від кількості відбитого предметом світла, що в свою чергу залежить від матеріала предмету. Так, наприклад, для виявлення предмету, виготовленого з гуми чорного кольору, необхідно наблизити його до датчику на відстань меншу в 5 разів, за ту відстань, якої вистачить для виявлення білого листу бумаги. Тому для регулювання потенціометра за відносну величину береться відстань, необхідну для білого листа бумаги. Параметри перелічені в таблиці [].

|  |  |
| --- | --- |
| Напруга живлення | 3.3 – 5 В |
| Тип датчику | Дифузійний |
| Компаратор | LM393 |
| Відстань до виявлення перешкоди | 2 – 30 см |
| Ефективний кут виявлення перешкод | 35° |
| Габарити | 43 х 16 х 7 мм |

Таблиця – Технічні характеристики датчику YL-63

Вихідний пін датчику YL-63 підключаєтсья до TRIG піна модуля сканеру E1005. Коли датчик виявляє перешкоду, він видає високий рівень сигналу із вихідного піну. Тобто цей сигнал надходить до піна E1005, що відповідає за активацію модуля. Тепер сканер готовий до ідентифікації штрих-коду. Такий принцип роботи пристрою створений для того, щоб сканер штрих-коду не випромінював світло тоді, коли це непотрібно, а тільки тоді, коли користувачу потрібно зчитати штрих-код з упаковки товару.

Як було сказано раніше, пристрій для ідентифікації штрих-коду знаходиться біля кошику для сміття, або почіплений безпосередньо на сам кошик. Тож коли користувач збирається викинути у сміття упаковку, для того щоб сповістити систему автоматизації про те, що конкретний товар був викинутий, достатньо лише піднести упаковку до датчику перешкод. Датчик виявить перешкоду і активує роботу сканеру. Користувачу залишиться піднести штрих-код до випромінювання сканеру. Після успішного зчитування штрих-коду, користувач може викинути упаковку у кошик, знаючи, що в системі моніторингу вже застосовані зміни, і товар більше не вважається наявним у житлі користувача.

Але для того, щоб виконати дії із програмної сторони системи, необхідно зісканований ідентифікатор передати до веб-застосунку. Ця дія має бути виконаною через мережу Інтернет. Тобто сканеру штрих-коду необхідно мати зв’язок з зовнішним світом.

Для прототипування макету було вирішено використати розглянуту раніше плату Arduino Mega 2560. До послідовних портів RX та TX було під'єднано відповідно виходи TX та RX модуля сканеру штрих-коду. Для забезпечення зв'язку з мережею Інтернет до Arduino було під’єднано модуль Wi-Fi ESP8266, що може працювати як шилд та отримує керування від плати через спеціальні AT-команди. В результаті отриманий макет був занадто громіздким для того, щоб бути розміщеним у житлі людини як автономна одиниця. Оскільки модуль ESP8266 має внутрішню пам’ять та може бути запрограмований як самостійний мікроконтролер, було вирішено перенести на нього всі задачі, що до цього виконувала плата Arduino Mega.

ESP8266 – це Wi-Fi модуль, що виробляється компанією Espressif та представляється як високоінтегроване рішення системи на кристалі для інтернету речей [<https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf#page=6&zoom=100,0,113>]. Завдяки широким та самодостатнім можливостям взаємодії з Wi-Fi мережею, ESP8266 моде виступати як самостійний базовий компонент для проектування систем з автоматизації розумного дому, так і бути відомим пристроєм по відношенні до ведучого пристрою у вже існуючій більш складній системі. Коли ESP8266 використовується як основний компонент проекту та має в собі вшитий прорамний код, він запускається з флеш-пам’яті та починає викноувати команди передбачені програмою. В той же час він може виступати в якості Wi-Fi адаптеру для будь-якого мікроконтроллера та отримувати від нього команді за SPI або UART інтерфейсами. Як, наприклад, цей модуль може бути засобом доступу до Інтернету плати Arduino Mega.

В модуль вбудовано розширену версію 32-бітного процесора Tensilica L106 Diamond series та вбудовану SRAM пам’ять. Комунікація з зовнішніми пристроями відбувається через цифрові піни GPIO. Наявне SDK надає приклади програм для побудовти простих проектів з використанням даного модулю.

Модуль ESP8266 підтримує повний стек протоколів 802.11 b/g/n. При використанні протоколу 802.11 n швидкість передачі даних модулем може сягати 72 Мбіт/с. Модуль праціює на частоті хвиль 2.4 ГГц, що є недоліком в розрізі того, що все більше і більше пристроїв отримують підтримку роботи на частоті 5 ГГц. Це дозволяє зменшити трафік з одного каналу мережі Wi-Fi шляхом розподілення нагрузки між двома канали. Для отримання вищої якості зв’язку, до модуля може бути під’єднана зовнішня антена. :Живиться ESP8266 від постійного струму з напругою 3 - 3.6 В. Слід зауважити, що чіп є вкрай чутливим до виходу напруги за визначиний діапазон. Навіть незначне перевищення може призвести до пошкодження чіпу. Тому вкрай необхідно забезпечити модуль стабільним живленням з напругою 3.3 В для стабільної роботи.

Процесор Tensilica L106, яким оснащений модуль, має дуже низкий рівень споживання енергії та досягає тактової частоти 160 МГц. В модулі присутня операційна система RTOS (Real-Time Operating System). Вона дозволяє модулю виконувати інструкції, що містяться в програмі, розміщеної у флеш пам’яті. Процесор має наступні інтерфейси для зовнішньої взаємодії:

* Програмовані інтерфейси ОЗУ та ПЗУ (iBus), які можна підключити до пам'яті контролера, а також використовувати для доступу до флеш-пам’яті.
* Інтерфейс оперативної пам'яті даних (dBus), який може з'єднуватися з контролером пам'яті.
* Інтерфейс AHB, який можна використовувати для отимання даних з реєстру.

Реалізація модулю, як системи на кристалі об’єднує пам’ять контролеру та окремі блоки ма’яті, включаючи SRAM і ROM. Мікроконтролер може отримати доступ до цієї пам’яті, як вже було сказано, через інтерфейси iBus, dBus та AHB. Для програмування модулю, розробнику доступно до 16 Мб флеш-пам’яті

ESP8266, як і Arduino, існує в багатьох реалізаціях. Кожна з них відрізняється типом антени, кількістю виведених пінів від мікроконтролера назовню та кількістю флеш-пам’яті.

Для комунікації сканеру штрих-кодів та Wi-Fi модулю по UART інтерфейсу необхідно з’єднати піни RX та TX ESP8266 відповідно до TX та RX пінів E1005. Тепер декодований ідентифікатор буде надходити від сканеру штрих-коду до Wi-Fi модулю, де він має бути сформований в HTTP запит та выдправлений до API.

Найпрлостіша у використанні версія виконання модулю ESP8266 навзивається ESP-01 та має всі необхідні піни для встановлення комунікації з модулем E1005. Зовнішній вигляд модулю ESP-01 представлено на рисунку [].

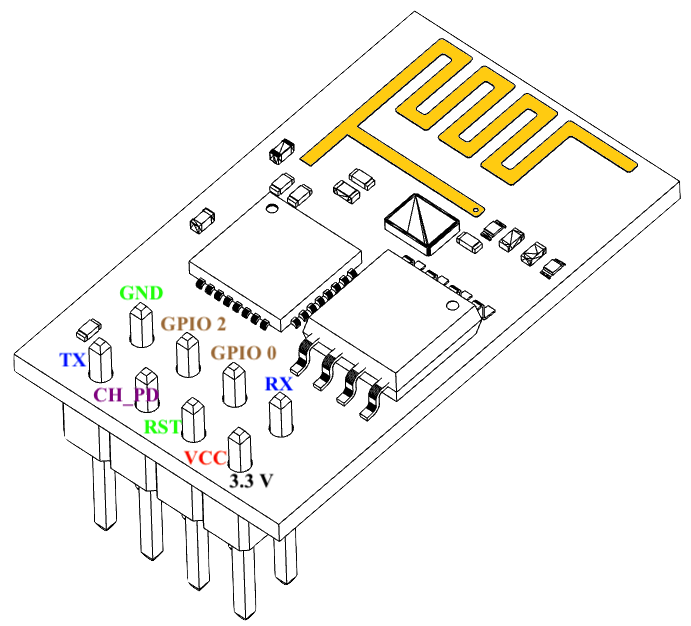


Рисунок – Зовнішній вигляд та піни модулю ESP-01 [<https://en.wikipedia.org/wiki/ESP8266#/media/File:ESP8266_01_PinOut.png>]

Це виконання модулю ESP8266 наразі є одним за найпоширеніших та має деякі недоліки стосовно більщ повноцінних реалізацій, що використовуються в інших пристроях розроблюваного апаратного комплексу. Можна перелічити наступні конструктивні недоліки ESP-01:

* На платі моделю розведено всього 8 пінів навідміну від 22 в інших реалізаціях
* Через те що більшість пінів не виведені на плату, під час запуску модуля вони знаходяться у невизначеному стані, що може вплинути на стабільність роботи модулю під час старту роботи
* Не підтримує режиму «глибокого сну»

Але всі недоліки, зазначені вище, не є вкрай важливими в контексті звичайного прийому та передачі даних. Вони ніяк не впливають на обчислювальні здібності процесору, тож не впливають на хід виконання програми. Також слід зазначити що модуль має найменші розміри серед інших реалізацій ESP8266 (14.3 х 24.8 х 3 мм), тож добре підходить до виконання поставлених задач. У таблиці [] наведено перелік всіх пінів модулю ESP-01.

|  |  |  |
| --- | --- | --- |
| № | Назва | Опис |
| 1 | GND | Заземленя |
| 2 | GPIO2 | Цифровий пін вводу/виводу |
| 3 | GPIO0 | Цифровий пін вводу/виводу |
| 4 | RXD | Прийом даних з ынтерфейсу UART0 |
| 5 | VCC | Живлення 3.3 В |
| 6 | RST | Зовнішній зброс налаштувань |
| 7 | CH\_PD | Активація чіпу |
| 8 | TXD | Передача даних з ынтерфейсу UART0 |