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

На правах рукопису

Кваліфікаційна робота на здобуття ступеня вищої освіти «бакалавр»

РОЗРОБЛЕННЯ ЗАХИЩЕНОЇ СИСТЕМИ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ БЛОКЧЕЙН, НАСКРІЗНОГО ШИФРУВАННЯ

Здобувача вищої освіти
Коломійця Володимира Валерійовича
Освітня програма
«Автоматизація та комп'ютерно-інтегровані технології»
(Спеціальність
174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка»)

_____
(підпис)

Науковий керівник:
д.т.н., професор Кисельов В.Б.

_____
(підпис)

Національна шкала_____
Кількість балів_____
Оцінка: ECTS

Київ – 2026

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

«ЗАТВЕРДЖУЮ»
Завідувач кафедри
_____Наталія ОМЕЦИНСЬКА
«____» _____2025_року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ
Коломійця Володимира Валерійовича

1. Тема роботи: Розроблення захищеної системи домашньої автоматизації з використанням технології блокчейн, наскрізного шифрування
керівник роботи: _д.т.н., професор Кисельов В.Б._____
затверджені Наказом ТНУ імені В.І Вернадського:
від «7 » листопада 2025 № 138/1-ОД

2. Строк подання студентом роботи: 25.05.2026
3. Вихідні дані до роботи: 1) Програмно-апаратний комплекс системи «Розумний будинок».

4. Зміст розрахунково-пояснювальної записки: Дослідити теоретично-методологічну базу систем домашньої автоматизації; Провести аналіз і виявити сучасні тенденції систем домашньої автоматизації; Запроектувати загальну архітектуру та розробити функціональний опис систем «Розумний будинок» та реалізувати прототип спроектованої системи; Пропрацювати охорону праці та безпеку в навколишньому середовищі._____

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

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ 1	професор Володимир Кисельов		
Розділ 2	професор Володимир Кисельов		
Розділ 3	професор Володимир Кисельов		
Розділ 4	професор Володимир Кисельов		

7. Дата видачі завдання _____ 2025 р.

КАЛЕНДАРНИЙ ПЛАН

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

ЗМІСТ

ВСТУП……07
РОЗДІЛ 1 СУЧАСНЕ УЯВЛЕННЯ СИСТЕМ ДОМАШНІЙ АВТОМАТИЗАЦІЇ «РОЗУМНИЙ БУДИНОК»……...……10
1.1 Загальна концепція систем «Розумного будинку»…….....11
1.2 Мережі і протоколи……13
1.3 Пристрої…….....14
1.4 Аналіз сучасного ринку інтелектуальних систем автоматизації житла…….16
Висновок до розділу 1……...21
РОЗДІЛ 2 РОЗРОБКА ПРОТОТИПУ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ «РОЗУМНИЙ БУДИНОК»……..23
2.1 Постановка задачі……....23
2.2 Загальна архітектура системи…….....24
2.3 Центральний пристрій…….....25
2.4 Кінцеві пристрої……..29
2.5 Мобільне застосування……...36
Висновок до розділу 2……..41
РОЗДІЛ 3 РОЗРОБКА МОДУЛЯ ГОЛОСОВОГО КЕРУВАННЯ ТА ОБРОБКИ ПРИРОДНЬОЇ МОВИ ДЛЯ СИСТЕМИ «РОЗУМНИЙ БУДИНОК»…….43
3.1 Розпізнавання запитів з параметрами……45
3.2 Розпізнавання запитів загального виду…….....48
3.3 Укладення по розпізнаванню семантики…….....51
3.4 Укладення по голосовому асистенту…….....52
Висновок до розділу 3……...53
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ 55
4.1 Опис робочого місця програміста 55
4.2 Освітленість робочого столу..... 58
4.3 Безпека в надзвичайних ситуаціях…….59
Висновок до розділу 4 61
ЗАГАЛЬНИЙ ВИСНОВОК..... 62
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 65

АНОТАЦІЯ

Коломієць В. В. Розроблення захищеної системи домашньої автоматизації з використанням технології блокчейн, наскрізного шифрування. – Рукопис.

Кваліфікаційна бакалаврська робота за спеціальністю 174 «Автоматизація, комп’ютерно-інтегровані технології та робототехніка». – Таврійський національний університет імені В.І. Вернадського, Київ, 2026 рік.

У бакалаврській кваліфікаційній роботі досліджено сучасні тенденції та теоретико-методологічну базу побудови бездротових сенсорних мереж для систем «Розумний будинок». Розроблено програмно-апаратний комплекс на основі децентралізованої архітектури, що використовує одноплатний комп'ютер Raspberry Pi 3 b+ у ролі центрального хаба та мікроконтролери ESP8266 для кінцевих пристроїв (датчики температури і вологості, датчики протікання, розумна розетка). Створено мобільний додаток для ОС Android із графічним інтерфейсом для управління пристроями, моніторингу їх стану та редагування сценаріїв автоматизації. Особливу увагу приділено створенню підсистеми обробки природної мови (NLP) для голосової взаємодії, із застосуванням контекстно-вільних граматик (Yargy парсер), векторного простору Word2Vec та морфологічного аналізатора PyMorphy2. Практичне значення полягає у створенні універсальної, масштабованої системи домашньої автоматизації класу Plug & Play, яка забезпечує високий рівень комфорту та безпеки.

Об’єкт дослідження – системи автоматизації житлових приміщень.

Предмет дослідження – архітектурні принципи та програмно-апаратні засоби побудови сучасних комплексів «Розумний будинок».

Мета роботи – проектування та практична реалізація програмно-апаратного комплексу, що інтегрує основні сучасні тенденції ринку домашньої автоматизації.

Методи дослідження – системний аналіз, апаратне та програмне прототипування, об’єктно-орієнтоване програмування, методи обробки природної мови (NLP), математичне моделювання.

Ключові слова: розумний будинок, інтернет речей (IoT), домашня автоматизація, Raspberry Pi, ESP8266, мобільний додаток, обробка природної мови, голосовий асистент.

ABSTRACT

Kolomiets V. V. Development of a secure home automation system using blockchain technology, end-to-end encryption. – Manuscript.

Qualification bachelor's thesis in specialty 174 "Automation, computer-integrated technologies and robotics". – Tavrichesky National University named after V.I. Vernadsky, Kyiv, 2026.

The bachelor's qualification work explores current trends and the theoretical and methodological basis for building wireless sensor networks for Smart Home systems. A hardware-software complex based on a decentralized architecture has been developed, which uses a Raspberry Pi 3 b+ single-board computer as a central hub and ESP8266 microcontrollers for end devices (temperature and humidity sensors, leak sensors, smart socket). A mobile application for Android OS with a graphical interface for managing devices, monitoring their status and editing automation scenarios has been created. Particular attention has been paid to the creation of a natural language processing (NLP) subsystem for voice interaction, using context-free grammars (Yargy parser), the Word2Vec vector space and the PyMorphy2 morphological analyzer. The practical significance lies in the creation of a universal, scalable Plug & Play home automation system that provides a high level of comfort and security.

The object of research is home automation systems.

The subject of research is architectural principles and software and hardware for building modern "Smart Home" complexes.

The purpose of the work is the design and practical implementation of a software and hardware complex that integrates the main modern trends in the home automation market.

Research methods – system analysis, hardware and software prototyping, object-oriented programming, natural language processing (NLP) methods, mathematical modeling.

Keywords: smart home, Internet of Things (IoT), home automation, Raspberry Pi, ESP8266, mobile application, natural language processing, voice assistant.

ВСТУП

Еволюційний шлях, який подолали інформаційні технології з моменту свого зародження у середині XX століття до сьогодні, є відносно коротким у контексті загальної історії людства. Проте саме цей період став визначальним для формування сучасного цифрового суспільства. Ключовим чинником такої стрімкої популярності IT-рішень виступила глобальна мережа Інтернет, яка забезпечила користувачам безперешкодний доступ до колосальних обсягів даних. Окрім інформаційної функції, вирішальне значення мав розвиток засобів миттєвої комунікації, що дозволили долати будь-які відстані в режимі реального часу.

Паралельно з розвитком засобів спілкування, поява глобальної мережі стимулювала інтерес інженерів та винахідників до концепції дистанційного моніторингу побутових приладів. Історичним прикладом такої ініціативи є демонстрація Джона Ромкі у 1990 році, який представив модель віддаленого управління тостером. Згодом, у 1999 році, для опису подібних систем почали використовувати термін «Інтернет речей» (IoT) [9]. За визначенням аналітиків Cisco, ера IoT фактично розпочалася тоді, коли кількість підключених до мережі інтелектуальних об'єктів перевищила загальну кількість людей на планеті [4].

На сучасному етапі технології Інтернету речей інтегровані у більшість галузей людської діяльності, включаючи промислову автоматизацію, медицину та сферу послуг, зокрема мережі швидкого харчування, як-от McDonald's [39]. Одним із найбільш динамічних сегментів застосування IoT є системи «Розумного будинку», що спрямовані на підвищення рівня комфорту, безпеки та енергоефективності приватного житла.

Сфера систем домашньої автоматизації на сьогоднішній день вийшла з фази раннього формування та перейшла до етапу масового впровадження. Якщо раніше ринок наповнювали переважно локальні стартапи, то зараз основний вектор розвитку задають глобальні технологічні екосистеми від Google (Google Home), Apple (Apple HomeKit), Amazon (Alexa) та Xiaomi (Mi

Home). Особливо важливим етапом став 2024 рік, коли впровадження єдиного стандарту сумісності Matter дозволило пристроям від різних виробників безперешкодно взаємодіяти між собою, що раніше було головною перепоною для користувачів.

За даними аналітичних звітів (зокрема Grand View Research та Fortune Business Insights), обсяг світового ринку розумних будинків у 2024 році перевищив 125 мільярдів доларів. Станом на 2025-2026 роки спостерігається ще стрімкіша динаміка: ринок оцінюється у понад 150–180 мільярдів доларів з прогнозованим зростанням до 400 мільярдів до 2030 року. Середньорічний темп приросту (CAGR) наразі становить близько 21–27%, що значно перевищує показники минулих років. Це підтверджує, що споживчий інтерес не просто зберігається, а масштабується завдяки доступності компонентів та розвитку штучного інтелекту.

В Україні, попри складні умови, також спостерігається попит на системи автоматизації, особливо в сегментах енергоефективності та безпеки. Зростання популярності пристроїв для моніторингу енергоспоживання та автономних систем захисту житла робить розробку вітчизняних програмно-апаратних комплексів критично актуальною. Таким чином, розробка інтелектуальної системи, що розглядається в межах даної бакалаврської роботи, відповідає не лише загальносвітовим тенденціям, а й специфічним потребам сучасного локального ринку.

Об'єктом даної бакалаврської роботи є системи автоматизації житлових приміщень. Предметом дослідження виступають архітектурні принципи та програмно-апаратні засоби побудови сучасних комплексів «Розумний будинок». Метою роботи є проектування та практична реалізація програмно-апаратного комплексу, що інтегрує основні сучасні тенденції ринку домашньої автоматизації.

Для досягнення поставленої мети в роботі було вирішено такі завдання:

1. Дослідити теоретичну базу та методологічні підходи до побудови систем автоматизації.

2. Проаналізувати поточний стан ринку та виявити ключові технологічні

тренди.

3. Розробити загальну архітектуру та детальний функціональний опис системи «Розумний будинок».

4. Створити та протестувати прототип спроектованого комплексу.

Теоретичний фундамент дослідження базується на роботах відомих фахівців у галузі IoT та систем автоматизації, таких як Р. Краненбург, Г. Лобаккаро, А.А. Попов, Д.В. Байгозін, А.Е. Китаєв, М.А. Шнепс-Шнеппе та Я.Ю. Григор'єв. Додатково було вивчено питання людино-машинної взаємодії (Я. Нільсен, Б. Шнейдерман, Д. Раскін) та методи обробки природної мови (Н. Хомський, М. Коробов).

Структура бакалаврської роботи обумовлена логікою дослідження і складається з п'яти розділів, які охоплюють як теоретичні аспекти, так і практичні результати розробки. Під час виконання роботи було ідентифіковано ключові компоненти сучасних систем (центральний хаб, кінцеві модулі, мобільний додаток та голосове управління), спроектовано архітектуру та успішно реалізовано комплект прототипів пристроїв разом із відповідним програмним забезпеченням.

РОЗДІЛ 1 СУЧАСНЕ УЯВЛЕННЯ СИСТЕМ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ «РОЗУМНИЙ БУДИНОК»

Нині, термін інтернет речей став досить популярним і широко використовується, як в публіцистиці, так і в наукових роботах. Першим використанням цього терміну прийнято вважати виступ Кевина Эштона на презентації для керівництва Procter & Gamble в 1999 році. На цьому виступі під інтернетом речей пропонувалося розуміти систему радіочастотних міток, націлену на управління логістичними ланцюгами в корпорації [28]. Наступним ключовим моментом в історії становлення інтернету речей є 2008 - 2009 року, коли стався перехід від «Інтернету людей» до «Інтернету речей», тобто кількість підключених до мережі предметів перевищила кількість людей.

Враховуючи таке стрімке зростання кількості різноманітних предметів, які підключені до мережі інтернет, дивним є той факт, що чітко сформульованого і широко визнаного визначення у терміну інтернету речей так і не з'явилося. У цій роботі під інтернетом речей розумітиметься «концепція простору, в якому все з аналогового і цифрового світів може бути поєднано, - це перевизначить наші стосунки з об'єктами, а також властивості і суть самих об'єктів», - це визначення дав керівник проекту розвитку IoT у рамках програми ЄС, Роб Ван Краненбург [41].

Спираючись на наведене вище визначення, можна з упевненістю судити про те, що інтернет речей не закінчується на окремо взятому об'єкті або групі об'єктів, але також включає їх статичне або динамічне об'єднання шляхом дротяних або безпровідних каналів зв'язку, підключених до мережі інтернет.

Слід уточнити концепцію «речі». Під «речами» найчастіше розуміють фізичні об'єкти, запрограмовані на обмін інформацією один з одним або із зовнішнім середовищем.

Проте навіть після того, як термін інтернету речей придбав чітке визначення, досить важко судити про те, що входить в це поняття. Згідно з Робом Ван Краненбургу IoT представляє з себе интернетсоединение чотирьох рівнів [7]:

1. Body area network(BAN) - рівень ідентифікація речей, як кожен окремий об'єкт, наприклад, розумний годинник, смарт-футболки;

2. Local area network(LAN) - споживчий рівень, як мережа персональних об'єктів, наприклад, «розумний будинок»;

3. Wide area network(WAN) - урбаністичний рівень, як акумуляція усієї інформації, яка відноситься до жителів окремого міста, наприклад, транспортні засоби;

4. Very wide area network(VWAN) - глобальний рівень, який охоплює інформацію, що стосується державних послуг, які можна отримати через інтернет, і вони більше не прив'язані до фізичного місцерозташування.

У цій роботі мова піде про другий рівень - LAN, а конкретно про системи типу «розумний будинок». Такі системи призначені для домашньої автоматизації певного ряду завдань без участі людини. А також надання інформації користувачеві про стан системи і довкілля, наприклад, про температуру і вологість повітря, стан пристроїв.

1.1 Загальна концепція систем розумного будинку

У більшості випадків системи розумного будинку є модульними і можуть бути різними варіантами компонувань пристроїв, які реалізують централізовану або децентрализованную систему. Основною відмінністю децентрализованной системи від централізованої є виконання самих пристроїв. А саме, кожен пристрій має свій власний обчислювальний модуль (мікропроцесор або мікроконтроллер) і свою власну енергонезалежну пам'ять. Це дозволяє забезпечити надійність таких систем, у разі виходу з ладу одного або декількох пристроїв, за винятком провідного пристрою.

Для розуміння структурної концепції систем розумного будинку розглянемо будову і призначення компонентів загального випадку архітектури децентралізованої системи (рисунок 1.1) [35]. Як правило такі системи складаються з хаба, хмари, мобільного застосування і набору пристроїв.

Так званий хаб є центральним пристроєм, який забезпечує об'єднання усіх

інших компонентів системи розумного будинку в єдину мережу. Реалізація такої мережі може бути як дротяною, так і безпровідною або їх комбінацією. Окрім з'єднання пристроїв, хаб виконує завдання по менеджменту компонентів системи, тобто додавання, видалення пристроїв, пересилка усіх повідомлень, що поступають в систему і в деяких випадках зберігання накопиченої інформації. Тобто хаб виступає в ролі диригента і управляє оркестром пристроїв, на основі команд користувача або заздалегідь визначених сценаріїв.

Набір кінцевих пристроїв — це одна найважливіших частин будь-якої системи розумного будинку. Кінцеві пристрої виконують різні завдання, які можна класифікувати на два основні напрями, а саме збір даних про навколишній світ і робота впливу на навколишній світ. Детальніший опис пристроїв, які можуть бути присутніми в системі розумного будинку будуть описані в пункті 3 розділу 1.

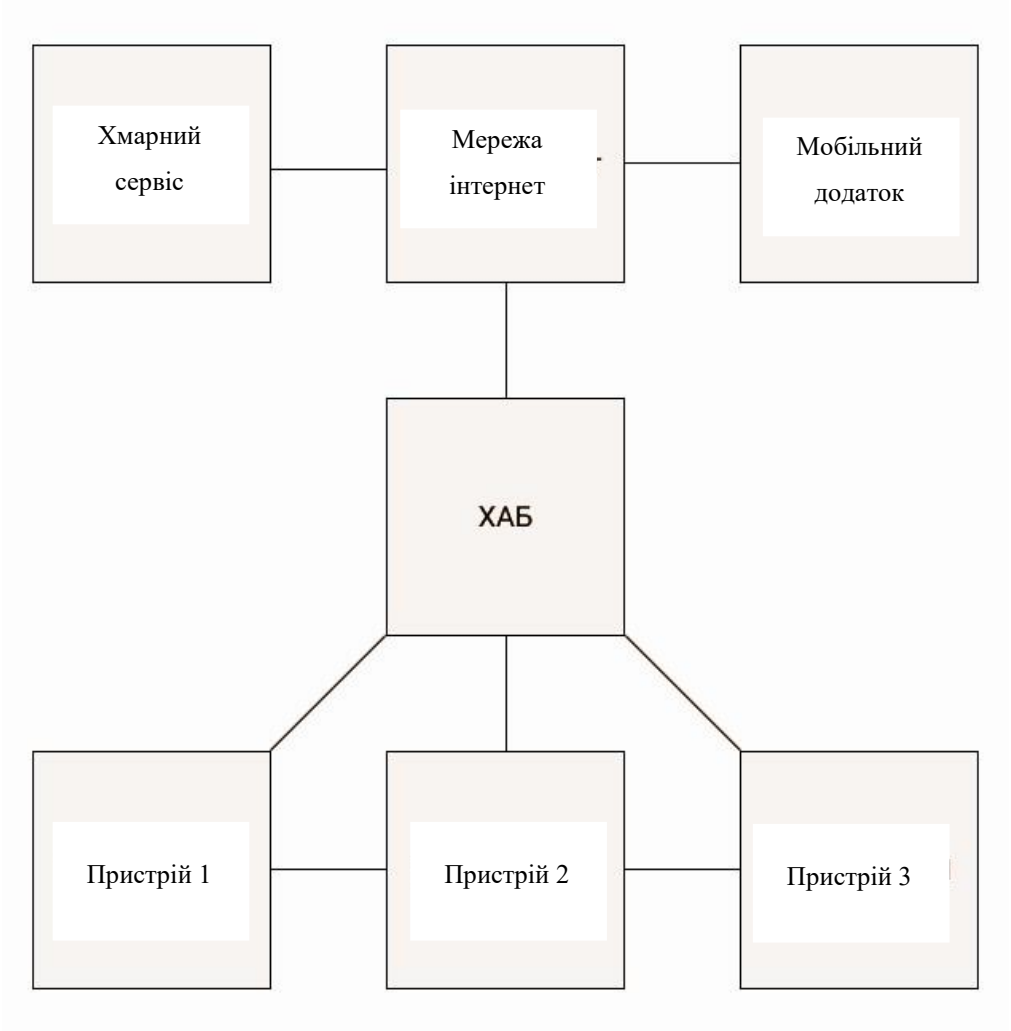


Рисунок 1.1 - Загальна архітектура децентрализованной системи розумного будинку

Інша поширена частина системи розумного будинку - мобільне застосування або будь-який інший інтерфейс взаємодії з системою. Найчастіше інтерфейс представляється у вигляді графічного призначеного для користувача інтерфейсу (англ. graphical user interface, GUI), наприклад, web або мобільне застосування. Основні завдання цього компонента охоплюють функціонал по наданню користувачеві зібраної інформації про стан довкілля і стан системи. Також за допомогою мобільного застосування користувач може віддавати команди системі.

Окремо варто виділити хмарні технології, які рідше зустрічаються в системах розумного будинку, але набирають популярність і можуть значно розширити функціонал системи. Мета «хмар» полягає у зберіганні і обробці даних. Це викликано тим, що локальні апаратні ресурси часто не мають достатньої продуктивності для виконання тривалих і ресурсоємних завдань. Окрім цього, «хмари» можуть виконувати роль видаленого сервера, який дозволяє користувачеві зв'язуватися з локальною системою розумного будинку віддалено, наприклад, за допомогою мобільного або web застосувань.

1.2 Мережі і протоколи

Для підключення пристроїв до системи розумного будинку потрібна реалізація дротяних або безпровідних мереж. Реалізація того або іншого типу мережі обумовлено безліччю різноманітних чинників: від завдання, що вирішується пристроєм, до енергоспоживання.

Дротяні мережі реалізується на основі безпосереднього з'єднання одного пристрою з іншим за допомогою кабельної проводки. Це дозволяє забезпечити надійний і стабільний канал зв'язку. Реалізація такої мережі має низькі вимоги до продуктивності пристроїв в порівнянні з безпровідними мережами. Дозволяє не прокладати додатковий канал для живлення пристроїв. Проте через наявності необхідності в прокладення каналів зв'язку, реалізація такої мережі переважна у будинку, що будується, квартирі. Ще одним обмеженням такої мережі для систем розумного будинку - вимога до достатньої кількості

портів у центрального пристрою, щоб забезпечити підключення усіх необхідних пристроїв.

Безпровідні мережі реалізуються на основі радіохвиль, що не вимагає безпосереднього фізичного з'єднання між пристроями. Це дозволяє не турбується про прокладення каналів зв'язку, що підходить для створення мережі у вже готовому приміщенні. Також такі мережі лояльніші до кількості пристроїв, що знаходяться в системі. Проте такий спосіб зв'язку менш стабільний зважаючи на велику кількість пристроїв, що знаходяться в ефірі.

Кожна мережа має набір своїх протоколів, правил, які забезпечують різноманітні способи передання даних, але накладають вимоги до підключення. У випадку з дротяними мережами вимоги відносяться як до продуктивності пристрою, так і до кількості портів введення\виведення, необхідних для встановлення з'єднання між пристроями. Для безпровідних облаштувань вимоги до апаратного забезпечення найяскравіше виражені по відношенню до продуктивності, а також до частоти радіоканалу, на якому здатний працювати пристрій. До дротяних протоколів відносяться: UART, I2C, 1 Wire, до безпровідних: WiFi, Bluetooth, Zigbee, RFID.

1.3 Пристрої

Автоматизація будь-якої системи розумного будинку базується на наявності кінцевих пристроїв, якими можна управляти за допомогою команд. Кінцеві пристрої охоплюють різноманітний спектр завдань. З визначення «речі» виходить, що кінцеві пристрої запрограмовані на обмін інформацією один з одним або із зовнішнім середовищем. Виходячи з цього визначення можна класифікувати усі кінцеві пристрої на два великі класи.

Пристрої, завдання яких зводиться до збору і\або зберіганню інформації, а також в деяких випадках і обробці цієї інформації, називаються датчиками. Як правило, такі пристрої є компактними, мають автономне живлення. Збирають один або декілька параметрів про довкілля. За параметрами, які збирає датчик, можна виділити наступні типи датчиків:

1) датчики температури і вологості;

2) датчики витоку води;

3) датчики вуглекислого газу;

4) датчики задимлення;

5) датчики відкриття\закриття дверей\вікон;

6) датчики присутності і датчики руху.

Це основний список датчиків, які найчастіше можна зустріти в системах розумного будинку. Вони забезпечують збір інформації про довкілля необхідної для вирішення завдання автоматизації.

Проте можуть зустрічатися і інші вузькоспеціалізовані датчики.

Другим класом кінцевих пристроїв є виконуючі пристрої. Їх основна мета не пов'язана зі збором інформації про навколишній світ, але може виконуватися як допоміжна. Основний функціонал таких пристроїв спрямований на вплив на навколишній світ, наприклад, підтримка мікроклімату шляхом регулювання термостата.

Окрім дистанційного або автоматичного керування, у функціонал виконуючих пристроїв може входити збір інформації про свій стан, яку згодом можна корисно використати. Прикладом такого використання інформації може послужити розумна розетка, яка здатна збирати інформацію про споживання електричного струму. Цю інформацію можна відображати користувачеві в початковому виді або обробляти і надавати виведення користувачеві, на основі чого користувач може знайти найбільш споживаючі пристрої і почати економити. Ще одним важливим прикладом збору інформації розумною розеткою є детектування різкого збільшення споживання, що може бути викликано виходом з ладу споживача. На основі цієї інформації пристрій відключить подання живлення.

Приклади виконуючих пристроїв :

1. Побутові пристрої: «розумний» чайник, посудомийка, пральна машина, холодильник;

2. Мультимедіа: smart TV, розумні колонки;

3. Системи створення мікроклімату: термостат, освіжувач повітря,

кондиціонер, тепла підлога.

1.4 Аналіз сучасного ринку інтелектуальних систем автоматизації житла

Аналіз тенденцій розвитку цифрових технологій свідчить про стрімку експансію засобів автоматизації у повсякденний побут. У 2024–2026 роках ринок систем «Розумний будинок» демонструє якісно нові показники порівняно з початковими етапами свого формування. Якщо звернутися до ретроспективного аналізу, то ще у 2018 році дослідження Global Smart Home Market Forecast від компанії Strategy Analytics вказували на те, що обсяг світового ринку досяг 84 мільярдів доларів, продемонструвавши 16% приріст. Проте станом на сьогоднішній день ці цифри значно зросли. За актуальними даними аналітиків IDC, обсяги постачань пристроїв для домашньої автоматизації на європейському ринку перевищують сотні мільйонів одиниць, що свідчить про стабільне щорічне зростання на рівні понад 20%. Особливо помітна активність у регіоні Центральної та Східної Європи, де темпи впровадження смарт-технологій випереджають середньосвітові показники.

Одним із головних чинників, що стимулюють популярність таких систем, стала масова цифровізація периферійних пристроїв, зокрема «розумних» акустичних систем та голосових асистентів. На початкових етапах розвитку ринку, як зазначав аналітик Ян Преносил, ключовим стримуючим фактором була відсутність підтримки локальних мов. Проте у 2024–2026 роках використання нейромережевих мовних моделей дозволило виробникам адаптувати інтерфейси для більшості регіонів, що зробило голосове керування пріоритетним способом взаємодії для масового споживача. Сучасна ринкова ситуація також характеризується чітким розподілом сфер впливу між глобальними корпораціями: у Азійсько-Тихоокеанському регіоні лідирують Xiaomi та LG, тоді як ринок Північної Америки та Європи розділений між Amazon, Google, Apple та Samsung.

Згідно з консолідованими прогнозами маркетингових агентств, таких як Discovery Research Group, світовий обсяг ринку систем «Розумний будинок» у

2026 році впевнено наближається до позначки у $200 мільярдів доларів. Очікувана динаміка зростання підтверджує, що інтелектуалізація житлового простору перетворилася з розкоші на стандарт сучасного будівництва. Візуалізація прогнозних показників, що відображає стабільний висхідний тренд, представлена нижче (Рисунок 1.2).

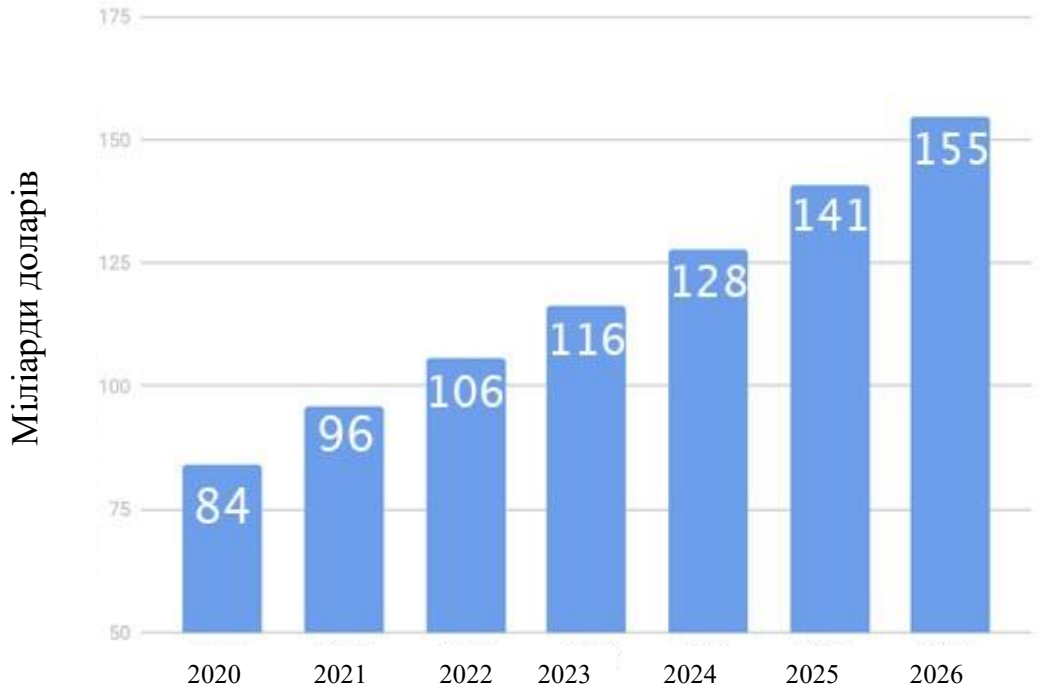


Рисунок 1.2 — Аналітичний прогноз динаміки капіталізації світового ринку Smart Home на довгострокову перспективу

Український сегмент ринку домашньої автоматизації також демонструє стійкість та поступове масштабування. Якщо у 2017–2018 роках обсяг ринку вимірювався сумою близько 9 мільярдів гривень, то в поточному періоді 2025–2026 років спостерігається значне збільшення інвестицій у цей сектор. Ринок структурно поділяється на три ключові компоненти: інтелектуальні кінцеві пристрої, сервісне обслуговування та професійні послуги з монтажу. Апаратна складова в Україні все ще значною мірою залежить від імпорту, що забезпечується постачанням продукції від таких гігантів, як Siemens, ABB, Samsung та Xiaomi. Водночас зростає частка вітчизняних стартапів, що спеціалізуються на хмарних рішеннях та розробці програмного забезпечення для контролерів.

Для детального розуміння структури пропозицій доцільно класифікувати

наявні рішення на три основні категорії. До першої належать системи «під ключ» — це складні індивідуальні проекти, що реалізуються такими компаніями як Schneider Electric або Schneider Electric. Вони відрізняються високою вартістю, але гарантують повну інтеграцію всіх інженерних мереж. Другу категорію складають системи типу «конструктор» (DIY), орієнтовані на технічних ентузіастів. Такі платформи, як HomeAssistant або Openhab, надають широкі можливості для кастомізації, проте вимагають від власника навичок програмування. Третя категорія — системи Plug & Play — є найбільш затребуваною завдяки простоті використання та доступній ціні.

У межах даного дослідження основна увага зосереджена саме на пристроях класу Plug & Play. До найвідоміших представників цього сегмента, що представлені на вітчизняному ринку, належать Rubetek, Apple HomeKit, Samsung SmartThings та Xiaomi Mi Home. Кожен із цих брендів пропонує власний погляд на концепцію розумного житла. Наприклад, компанія Rubetek акцентує увагу на бездротовій передачі даних та відсутності абонентської плати (Рисунок 1.3), тоді як Apple HomeKit та Samsung SmartThings забезпечують глибоку інтеграцію з мобільними екосистемами та підтримують широкий спектр сторонніх гаджетів (Рисунок 1.5, Рисунок 1.6). Система Mi Home від Xiaomi залишається лідером за кількістю доступних датчиків та ціновою привабливістю (Рисунок 1.7).



Рисунок 1.3 — Екосистема пристроїв для автоматизації житла від Rubetek



Рисунок 1.4 — Схема підключення компонентів у системі Life Control



Рисунок 1.5 — Інтерфейс мобільного додатка Apple Home для керування сценаріями



Рисунок 1.6 — Архітектура взаємодії пристроїв на базі Samsung SmartThings



Рисунок 1.7 — Модельний ряд інтелектуальних датчиків екосистеми Xiaomi Mi Home

Для об’єктивного порівняння функціональних можливостей зазначених систем було розроблено Таблицю 1. Вона відображає ключові відмінності в підходах до автоматизації, типах підтримуваних протоколів та можливостях голосової взаємодії. Аналіз показав, що попри різноманітність апаратних рішень, більшість виробників використовують схожі принципи побудови логіки керування, що ґрунтуються на сценаріях.

Таблиця 1

Зведений аналіз технічних характеристик популярних систем «Розумний будинок»

Параметр порівняння	Rubetek	Life Control	Home Kit	SmartThings	Mi Home
Пристрої	Власні + сторонні	Тільки власні	Переважно сторонні	Samsung + сторонні	Велика власна база
Додаток	Android / iOS	Android / iOS	Тільки iOS	Android / iOS	Android / iOS
Логіка роботи	Сценарії	Сценарії	Сценарії	Сценарії	Сценарії
Голосове керування	Сторонні сервіси	Немає	Siri	Bixby	Mi AI / Google

Підбиваючи підсумок проведеного аналізу ринку, можна стверджувати, що інтерес до технологій «Розумного будинку» постійно зростає як на глобальному, так і на локальному рівнях. Проте фрагментарність ринку та відсутність єдиних стандартів взаємодії часто створюють труднощі для кінцевого користувача. Саме тому спроектована у даній роботі система має

відповідати жорстким вимогам щодо універсальності. Вона повинна підтримувати три рівні взаємодії: фізичний контакт, графічний інтерфейс смартфона та голосові команди. Автоматизація процесів має відбуватися за допомогою гнучкої системи сценаріїв, де кожен тригер запускає визначений ланцюжок дій, забезпечуючи легку масштабованість та інтеграцію нових модулів.

Висновки до розділу 1

У першому розділі було проведено комплексне дослідження теоретико-методологічної бази та сучасного стану систем домашньої автоматизації. Системи «Розумний будинок» розглядаються як невід'ємна складова концепції Інтернету речей (IoT), що функціонує на споживчому рівні локальних мереж (LAN). Такі системи призначені для автоматизації побутових завдань без безпосередньої участі людини, а також для інформування користувача про стан довкілля. Визначено, що загальна архітектура децентралізованої системи переважно складається з центрального хаба (який відповідає за менеджмент мережі), хмарних сервісів (для зберігання та обробки даних), мобільного додатка (як графічного інтерфейсу користувача) та набору кінцевих пристроїв. Самі кінцеві пристрої концептуально поділяються на датчики для збору інформації (наприклад, температури чи руху) та виконуючі пристрої для безпосереднього впливу на середовище, такі як термостати чи розумні розетки.

Для надійної взаємодії всіх компонентів у системі можуть використовуватися як дротові (наприклад, UART, I2C, 1 Wire), так і бездротові (WiFi, Bluetooth, Zigbee, RFID) протоколи зв'язку. Бездротові мережі виявляються значно гнучкішими для інтеграції у вже побудовані та готові приміщення, тоді як дротові рішення забезпечують вищу стабільність каналу зв'язку та не потребують додаткового живлення радіомодулів. Аналіз сучасного ринку інтелектуальних систем автоматизації житла показав, що він демонструє стрімке зростання у 2024–2026 роках. Цьому сприяє загальна масова цифровізація, впровадження стандарту сумісності Matter, а також активне використання нейромережевих мовних моделей, що дозволило подолати

мовний бар'єр при голосовому управлінні. Серед трьох основних класів систем, які сьогодні представлені на ринку (індивідуальні дорогі розробки «під ключ», DIY-конструктори для технічних ентузіастів та системи Plug & Play), найбільш оптимальним і перспективним для масового сегмента є саме клас Plug & Play завдяки його простоті використання та загальній доступності.

Підбиваючи підсумки аналізу, можна стверджувати, що для подолання існуючої ринкової фрагментарності та відсутності єдиних стандартів взаємодії пристроїв, проектована в роботі система має бути максимально універсальною. Вона обов'язково повинна підтримувати три рівні взаємодії з користувачем: безпосередній фізичний контакт, дистанційне управління через графічний інтерфейс смартфона та подання голосових команд. В основі автоматизації процесів має лежати гнучка система сценаріїв, де кожен окремий тригер запускає визначений ланцюжок дій виконавчих пристроїв. Такий структурний підхід гарантує легку масштабованість системи та зручну інтеграцію нових модулів без необхідності повної перебудови мережі.

РОЗДІЛ 2 РОЗРОБКА ПРОТОТИПУ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ «РОЗУМНИЙ БУДИНОК».

2.1 Постановка завдання

Метою цієї роботи є розробка програмно-апаратного комплексу прототипу системи домашньої автоматизації «розумний будинок» відповідно до проведеному аналізом ринку і виявленими вимогами.

Така система націлена на облаштування житла людини за допомогою сучасних технологій для підвищення комфортності і зручності життя, забезпечення безпеки, раціонального використання ресурсів.

Дана система має бути децентралізованою, тобто кожен пристрій в системі має свій власний мікропроцесор або мікроконтроллер і постійну пам'ять. Що дозволяє відповідати наступним принципам:

1. Принцип модульності. Уся система складається з набору модулів, що несуть певний функціонал. Присутність або відсутність якого-небудь модуля не впливає на працездатність усієї системи, а тільки може змінити її функціональні можливості. Виключенням є хаб і маршрутизатор, які забезпечують мінімальний функціонал на всьому протязі роботи системи. Також мобільне застосування є обов'язковим компонентом, який має бути присутнім в системі, на момент первинного налаштування.

2. Принцип відмовостійкості. У разі виходу з ладу будь-якого пристрою, за винятком хаба і маршрутизатора, під час роботи системи може змінюватися тільки функціональним об'єм і не повинно відбиватися на стабільності і працездатності усієї системи в цілому.

Сучасні реалізації систем домашньої автоматизації мають різноманітний функціонал, який покриває потреби користувачів різних груп. Для розуміння основної концепції систем «Розумний будинок» скористаємося підходом по створенню мінімально життєздатного продукту (англ. Minimum viable product, MVP), тобто достатнім для розгортання і що має основні функціональні можливості.

Для комплексної системи «Розумний будинок» таким рішенням буде

програмно-апаратний комплекс, що охоплює усі аспекти подібних систем, виявлені при аналізі ринку систем домашньої автоматизації, включаючи вимоги, розглянуті в пункті 4.1 розділу 1.

Подібна система повинна реалізовувати наступні функціональні аспекти:

1. Організовувати локальну безпровідну мережу, до якої надалі підключатимуться кінцеві пристрої, центральний пристрій і мобільне застосування;
2. Забезпечувати менеджмент усіх компонентів в системі, включаючи динамічну зміну кількості компонентів і пересилку повідомлень між ними за певними правилами;
3. Зберігати і обробляти накопичену інформацію про стан системи;
4. Виконувати дії згідно із запитами користувача;
5. Реалізовувати доступ до системи на основі мобільного застосування;
6. Взаємодія за допомогою голосових команд на природній мові;
7. Автоматизація протікання заздалегідь запланованих алгоритмів на основі системи сценаріїв, спираючись на показники датчиків і події, що виникають в системі під час функціонування.

2.2 Загальна архітектура системи

Грунтуючись на наведених вище концептуальному і функціональному описі системи була розроблена загальна архітектура системи (рисунок 2.1).



Рисунок 2.1 - Загальна архітектура системи домашньої автоматизації «розумний будинок»

Прототип системи домашньої автоматизації, що розробляється, повинен відповідати основному функціональному охопленню таких систем. Для цього потрібна наявність наступних компонентів системи

1. Центральний пристрій
2. Набір прототипів кінцевих пристроїв
3. Мобільне застосування
4. Маршрутизатор

2.3 Центральний пристрій

Система домашньої автоматизації, що розробляється, «розумний будинок» складається з безлічі облаштувань різного призначення. Для об'єднання усіх

пристроїв в єдину мережу, що функціонує за певними правилами, потрібне деяке облаштування управління. Таким пристроєм є хаб, яке реалізують функцію менеджменту пулу пристроїв, пересилку повідомлень між ними, збір і зберігання даних.

Для реалізації центрального пристрою необхідно визначитися з основними характеристиками, властивими подібному класу пристроїв. Різні варіації систем розумного будинку по-різному інтерпретують облаштування управління. Для деяких систем цей пристрій прихований від очей, в деякому спеціальному технічному приміщенні. Інші виробники реалізують пристрій, що управляє, у вигляді компактного рішення, яке знаходиться у усіх на видноті. Звідси витікає вимога до компактності пристрою.

Зважаючи на компактність пристрою виникає питання про продуктивність цього рішення. Подібне рішення повинне та здатне обробляти одночасне з'єднання з декількома кінцевими пристроями, забезпечувати своєчасне виконання команд користувача, збирати і зберігати дані і на основі певних правил автоматично управляти системою. Виходячи з компактності пристрою можна розглянути пристрій, грунтований на мікроконтроллері, проте такі пристрої мають низьку продуктивність, недостатню для поставленого завдання. Рішенням є використання одноплатних комп'ютерів, які представлені на ринку різноманітною множиною під різні завдання. Найбільш популярним і доступним є Raspberry Pi 3 b+ (рисунок 2.2).

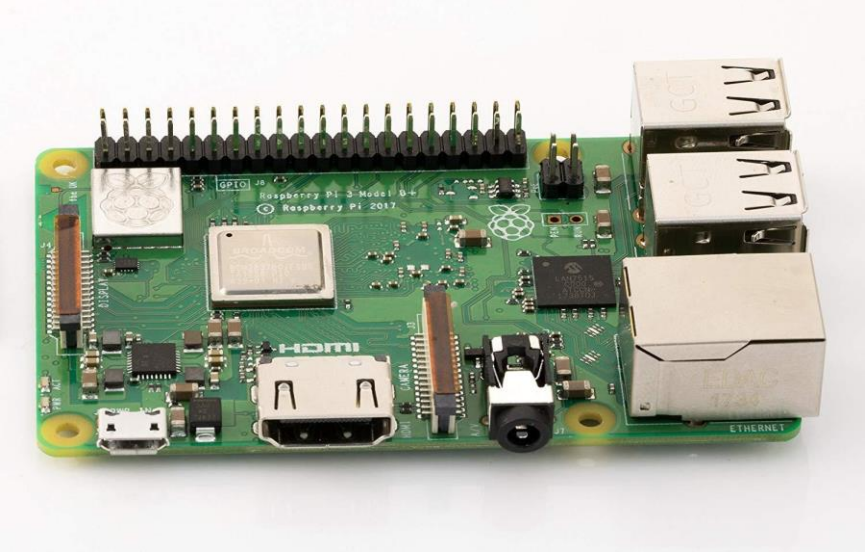


Рисунок 2.2 - Одноплатний комп'ютер Raspberry Pi 3 b+

Raspberry Pi 3 b+ — одноплатний комп'ютер, розроблений компанією Raspberry Pi Foundation. Має невеликі габарити 85x56x17 мм, що порівнянно з розмірами кредитної картки і задовольняє заявленій вимозі про компактність. Продуктивності таке рішення також буде досить для забезпечення багатопоточного виконання програми з вищеописаним функціоналом. Це можливо завдяки 64-бітовому 4-ядерному мікропроцесору Cortex — A53 на архітектурі ARM, тактова частота складає 1.4 Ггц, виконаний на однокристальному чіпі Broadcom BCM2837. Оперативна пам'ять представлена у вигляді 1Гб LPDDR2 SDRAM. Власна постійна пам'ять відсутня, проте є слот під карту пам'яті формату MicroSD.

Комунікація представлена у вигляді WiFi 802.11n, гигабитного RJ45 і Bluetooth 4.2 з підтримкою Bluetooth Low Energy. На додаток має достатню кількість портів введення\виведення, потрібного середньостатистичному користувачеві цього пристрою у вигляді HDMI, чотирьох USB 2.0 і композитного виходу 3.5 мм Але для реалізації центрального облаштування системи домашньої автоматизації це має низьке значення, на відміну від 40 портів введення\виведення загального призначення, які можна використати для підключення датчиків різного роду. Головною перевагою, виділяючим Raspberry Pi 3 b+, серед усіх інших — велике співтовариство, яке накопило велику кількість підходів і рішень до того або іншого завдання.

Вибраний одноплатний комп'ютер покриває закладені вимоги до апаратних характеристик пристрою, що розробляється. Цей комп'ютер дозволяє вести розробку програмного забезпечення на усіх популярних мовах програмування, включаючи такі мови як C\C++, Java, Python. Виходячи з поставленого завдання і обмеження за часом на розробку програмного забезпечення була вибрана мова програмування Python 3, який дозволяє досить швидко розробляти прототип поставленого завдання. Має велике співтовариство, яке розробило і поширює у відкритому доступі безліч різних модулів, які прискорюють розробку. Для розробки було вибрано середовище розробки PyCharm від компанії JetBrains.

Центральний пристрій є головним компонентом системи, яке забезпечує

контроль над усіма іншими пристроями. Увесь функціонал можна розбити на три великі модулі: взаємодія з пристроями, автоматизація протікання процесів і взаємодія з мобільним застосуванням.

Основа будь-якої системи домашньої автоматизації - кінцеві пристрої. Взаємодія з ними реалізується за допомогою менеджера пристроїв. На основі проведеного аналізу сучасних систем «Розумний будинок» було виявлено, що підключення до пристрою проходить в дві стадії. Спочатку необхідно підключитися до пристрою і передати інформацію по підключенню до локальної мережі. Після чого кінцевий пристрій підключиться до локальної мережі і переходить в режим штатного функціонування.

Raspberry Pi 3 має один модуль безпровідного зв'язку WiFi. Для проведення процедури ініціалізації потрібно буде відключитися від локальної мережі і підключитися до нового кінцевого пристрою. Це призводить до розриву з'єднання між центральним пристроєм і усіма іншими пристроями в системі. Тому для ініціалізації пристроїв необхідно додати ще один модуль безпровідного зв'язку WiFi. Цей модуль(далі WiFi 1) був реалізований на основі мікроконтроллера ESP8266 від китайського виробника Espressif(рисунок 2.3). Підключення Raspberry Pi 3 до ESP8266 організовано на основі UART.

Після додавання пристрою в систему необхідно проводити його обробку, а саме організувати передачу повідомлень між системою і кінцевим пристроєм. Усі повідомлення можна розділити на два типи: команди і дані. Притому команди вирушають тільки з центрального пристрою, а дані - з кінцевого.



Рисунок 2.3 - Мікроконтроллер ESP8266

Реалізація автоматизації представлено у вигляді менеджера сценаріїв. Кожен сценарій складається з двох основних компонентів: тригера і послідовності подій. Менеджер сценаріїв відстежує стан системи і звіряє його з умовою тригера. При виконанні умови виконується алгоритм послідовності подій, тобто передаються команди на відповідні пристрої.

Підключення мобільного додатка до центрального пристрою реалізоване в концепції архітектури клієнт - сервер. Сервером виступає центральний пристрій, в якості клієнта - мобільне застосування. Увесь період підключення мобільного застосування можна розбити на два типи з'єднання : пасивне і активне.

Під час взаємодії користувача з мобільним застосуванням мається на увазі активний тип з'єднання. У такому разі система має бути чуйною і миттєво реагувати на команди користувача. Для цього реалізований сервер на основі сокетного підключення, що забезпечує постійне з'єднання з пристроєм і мінімізує затримки.

Коли додаток є неактивним, тобто виконується у фоні, мається на увазі пасивний тип підключення. У цій ситуації системі немає необхідності у високій частоті оновлення інформації на мобільному застосуванні. Досить з деякою частотою опитувати систему про зміни. Для цього реалізований сервер на основі REST, що призводить до меншого енергоспоживання, ніж сервер на основі сокетного підключення.

2.4 Кінцеві пристрої

Ринок «розумних» пристроїв для систем домашньої автоматизації наповнений різноманітними пристроями. Притому пристрої, що мають схожий по функціональному охопленням, можуть бути реалізовані абсолютно різним чином. Це викликано як конкуренцією, так і відсутністю узгодженості в розробці.

Згідно з розділенням систем домашньої автоматизації на централізовані і децентрализованные, кінцеві пристрої можуть бути реалізовані як на основі мікроконтроллера, так і без його використання. Поточна ситуація на ринку

«розумних» будинків склалася таким чином, що для систем типу Plug & Play пристрої представлені у вигляді окремого, в деяких ситуаціях самодостатнього, пристрою. Тобто пристрою, який базується на окремому мікроконтроллері з підтримкою певного безпровідного стандарту зв'язку. Це викликано двома причинами:

1. Як правило, систему домашньої автоматизації придбавають вже для готового будинку\квартири і використання кабельної проводки неможлива або украй небажана. Вибір реалізації пристроїв на основі безпровідного з'єднання тягне необхідність у використанні мікроконтроллера.

2. Зважаючи на комплексність повноцінної системи домашньої автоматизації, рішення, яке може охопити увесь спектр потреб користувачів є витратним для середньостатистичного користувача. Це призводить до ситуації, коли користувач бажає придбати подібний товар, але не готовий купувати повноцінну систему. Таким чином з'являється необхідність, щоб пристрій міг функціонувати без використання центрального пристрою. Уся взаємодія відбувається безпосередньо з мобільним застосуванням.

Таким чином для реалізації прототипів пристроїв необхідно визначитися з мікроконтролером і мережевим стандартом, на основі яких буде проводиться розробка. На даний момент ринок мікроелектроніки дозволяє придбати мікроконтроллери для вирішення більшості завдань, які підходять для реалізації облаштувань різної міри складності.

Якщо звернути увагу на співтовариства по розробці пристроїв на основі мікроконтролерів, то найчастіше в якості основи використовую платформу Arduino. Ця платформа складається з програмної і апаратної частини. Програмна частина представлена у вигляді середовища розробки, апаратна - у вигляді плати з мікроконтроллером і обв'язуванням. Класична лінійка пристроїв базується на мікроконтроллерах Atmel AVR. Завдяки простоті використання і невеликій ціні плата отримала популярність. У зв'язку з цим виробники компонентів розробляють безліч різноманітної периферії що мають апаратну сумісність з цією платформою. Програмне забезпечення розробляється як виробниками пристроїв, так і співтовариством, що

сформувалося.

Проте через необхідності у безпровідному каналі зв'язку варто звернути увагу на рішення у вигляді мікроконтроллера, що має за умовчанням апаратну і програмну підтримку якого-небудь безпровідного стандарту зв'язку. Відповідно до запропонованої загальної архітектури системи «розумного» будинку (рисунок 2.1) в якості безпровідного стандарту зв'язку був вибраний WiFi. Безпровідний канал зв'язку, що реалізовується на основі цього протоколу достатній практично для будь-якого кінцевого пристрою, що в належній мірі дозволяє розробляти прототипи пристроїв. Проте варто відмітити, що для більшості пристроїв цей протокол є надмірним. А для пристроїв, що використовують автономне джерело живлення, використання цього стандарту безпровідного зв'язку недоцільно, зважаючи на існування інших стандартів менш вимогливих до енергоспоживання, наприклад, ZigBee, Bluetooth Low Energy.

В якості такого мікроконтроллера був вибраний, вже згаданий, ESP8266 (рисунок 2.2). Цей мікроконтроллер володіє 80 MHz 32 - bit мікропроцесором Xtensa - Tensilica L106 і підтримує безпровідні стандарти IEEE 802.11 b/g/n. Має 14 портів введення\виведення загального призначення, які буде досить для створення прототипів нескладних пристроїв. Живлення здійснюється в діапазоні від 2.2 до 3.6 вольт, що відповідає більшості пристроїв, сумісних з платформою Arduino.

В якості прототипів кінцевих пристроїв для системи, що розробляється, було реалізовано два датчики і одно виконуючий пристрій. Кожен пристрій грунтований на мікроконтроллері ESP8266. Для проектування принципових схем підключення компонентів використовувалася EasyEDA. Програмне забезпечення пристрою розділене на дві складові:

1. Прошивка мікроконтроллера. Цей програмний код призначений для управління мікроконтроллером і периферією пристрою. Для розробки використовувалося середовище Arduino.

2. Модуль для центрального пристрою. Потрібний для уніфікованої взаємодії з усіма пристроями. А також дозволяє робити обчислення за рахунок

засобів центрального пристрою.

2.4.1 Прототип датчика температури і вологості

Один з найпоширеніших пристроїв в системі домашньої автоматизації - датчик температури і вологості, який дозволяє відстежувати поточні значення температури і вологості довкілля. Окрім простого інформування, ці значення можна використати для автоматизації підтримки мікроклімату.

Основою такого пристрою був вибраний датчик DHT11(рисунок 2.4), який часто використовується для створення метеостанцій власної розробки або для впровадження в системи розумного будинку. Цей датчик має невеликі габарити: 15.5x12x5.5 мм Живлення датчика підтримується в діапазоні від 3 до 5 вольт, що відповідає живленню і логічним рівням ESP8266 і не зажадає додаткового перетворення напруги. Діапазон виміру температури складає від 0 до 50 градусів Цельсія з погрішністю в 2%. Дані показників вистачає для створення прототипу облаштування виміру температури і вологості.

Для реалізації апаратної частини цього пристрою необхідно використати наступні компоненти: ESP8266, DHT11 і автономне джерело живлення 3.3 вольта. На основі цих компонентів була розроблена принципова схема підключення компонентів (рисунок 2.5).

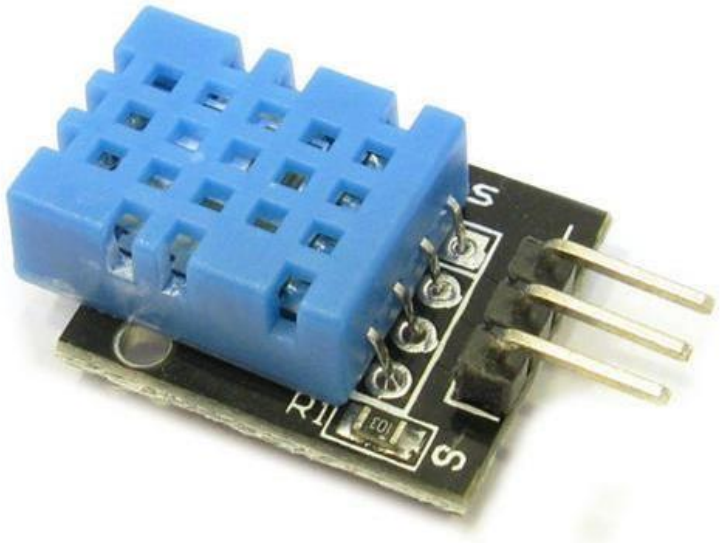


Рисунок 2.4 - Датчик температури і вологості DHT11

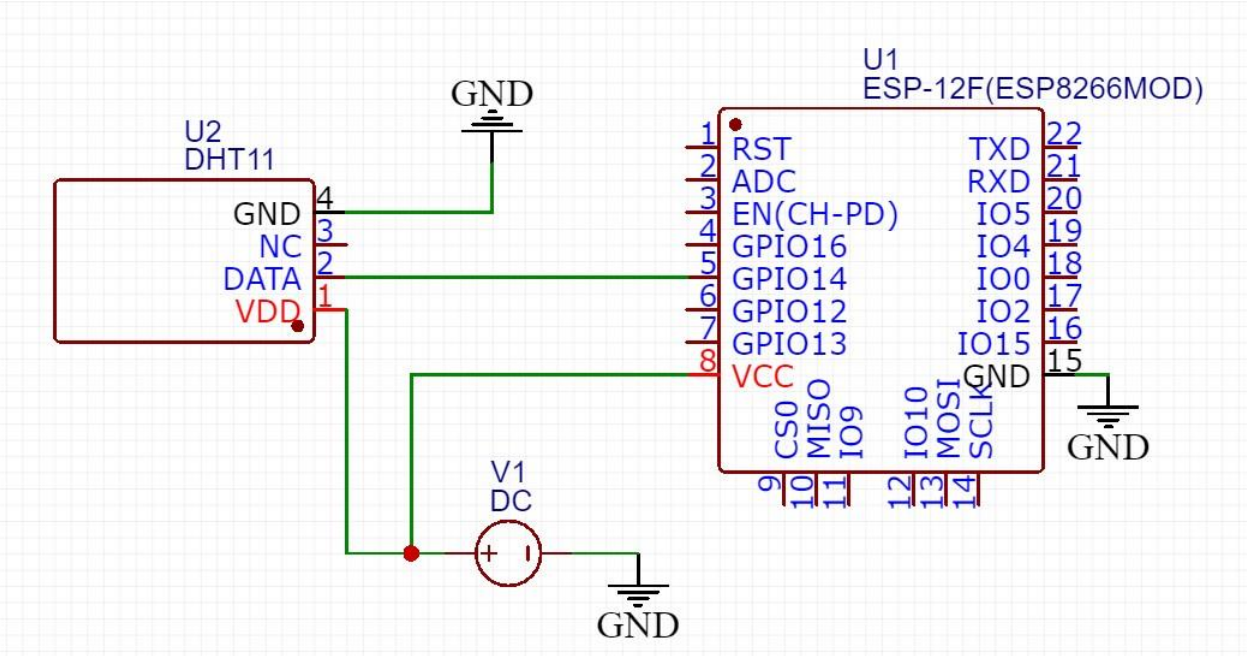


Рисунок 2.5 - Принципова схема прототипу датчика температури і вологості

Загальний алгоритм пристрою розділений на дві стадії. Спочатку ESP8266 знаходиться в стані очікування ініціалізації. Для цього WiFi переведений в стан точки доступу і створений сервер, який чекає повідомлення від підключених клієнтів у форматі JSON. Коли користувач додасть новий пристрій, центральний пристрій за допомогою модуля WiFi 1 підключиться до цього пристрою і відправить ім'я і пароль локальної мережі. Після цього пристрій перейде в режим крайової станції і підключиться до локальної мережі. Надалі пристрій відправлятиме повідомлення на центральний пристрій у форматі JSON, в які включені значення температури і вологості наступного виду: {"TEMPERATURE": 22.1, "HUMIDITY": 52}. Подальша обробка і зберігання даних проходить на центральному пристрої.

2.4.2 Прототип датчика протікання

Існує безліч різних датчиків, призначення яких - підвищення безпеки. Одним з таких пристроїв є датчик протікання. Його єдине завдання - інформувати користувача про виявлені протікання. Цей пристрій буде корисний в місцях наявності води, наприклад, ванні кімнати і кухня.

Розробка такого пристрою розпочинається з визначення датчика, який відстежуватиме наявність протікання. Поширеним датчиком, який

застосовують для створення систем домашньої автоматизації власної розробки, є Water Sensor (рисунок 2.6). Основне призначення датчика - визначення рівня води, який також дозволить детектувати її виявлення. Живлення датчика можливе в діапазоні від 3.3 до 5 вольт, розміри складають 62x20x8 мм



Рисунок 2.6 - Датчик рівня води Water Sensor

Компоненти і схема підключення схожа з датчиком температури і вологості, виключенням є необхідність в підключенні до аналогового входу ESP8266, оскільки датчик має аналогових вихід (рисунок 2.7).

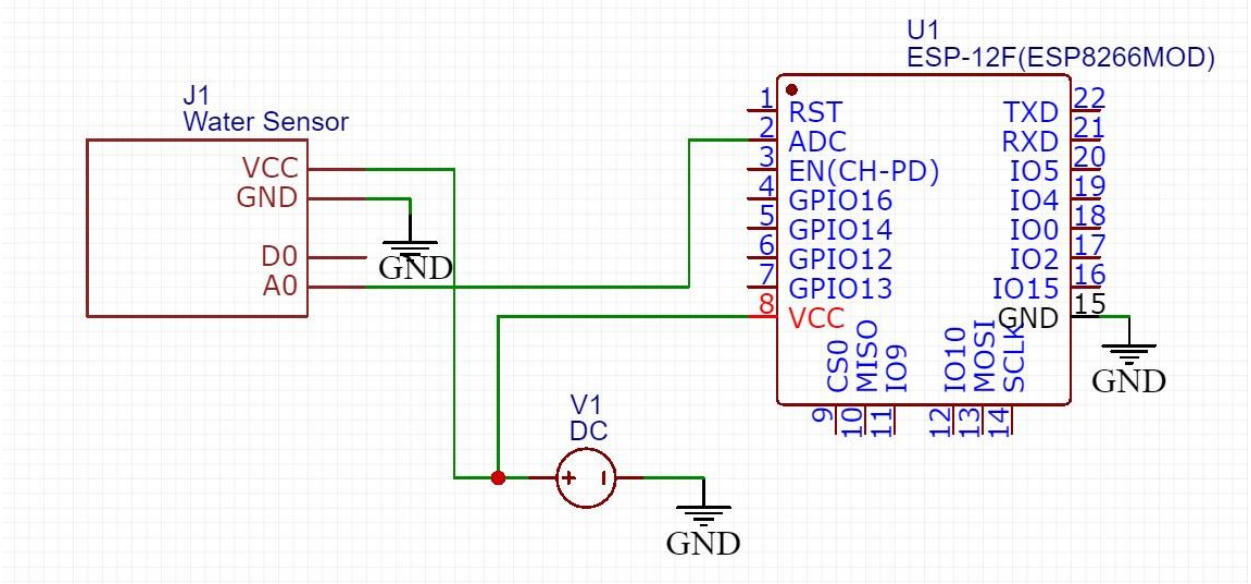


Рисунок 2.7 - Принципова схема прототипу датчика протікань

Загальний алгоритм роботи схожий з прототипом датчика температури і вологості. Відрізняються виключно дані, які відправляє пристрій. Вони містять значення рівня води у виді {"WATER": 0}. Свідчення рівня води може набувати значень в діапазоні від 0 до 1023, що викликано розрядність АЦП

ESP8266. Проте для прототипу, що розробляється, така точність навіть зайва, спрацьовування датчика відбувається при досягнення порогового рівня.

2.4.3 Прототип розумної розетки

Одним із вже класичних виконуючих облаштувань домашньої автоматизації, нарівні з розумною лампочкою, є розумна розетка. Основним функціоналом таке пристрої являється дистанційне включення і відключення подання живлення на навантаження, тобто пристрій, який підключений до розетки. Додатково такі пристрої здатні відстежувати споживання енергії і показники. Для створення прототипу «розумної» розетки був вибраний основний функціонал - дистанційне включення\відключення подання живлення на навантаження.

Для реалізації такого пристрою необхідно скористатися реле, яке здатне управлятися за допомогою ESP8266. Для цього використовуємо електромеханічним реле SRD - 05VDC - SL – C (рисунок 2.8). Це реле здатне працювати як зі змінною напругою до 250 вольт при струмі до 10 ампер, так і з постійною напругою до 28 вольт при струмі до 10 ампер. Що дозволяє реалізувати прототип розумної розетки.



Рисунок 2.8 - Електромеханічне реле SRD - 05VDC - SL - C

Реалізація цього пристрою значно відрізняється від попередніх датчиків. Основна відмінність викликана наявністю змінної напруги в мережі. Для цього пристрою недоцільно використати автономне джерело живлення, тому необхідно додати блок живлення, який перетворюватиме змінну напругу в

постійну. У відповідність з цим була розроблена принципова схема прототипу розумної розетки (рисунок 2.9).

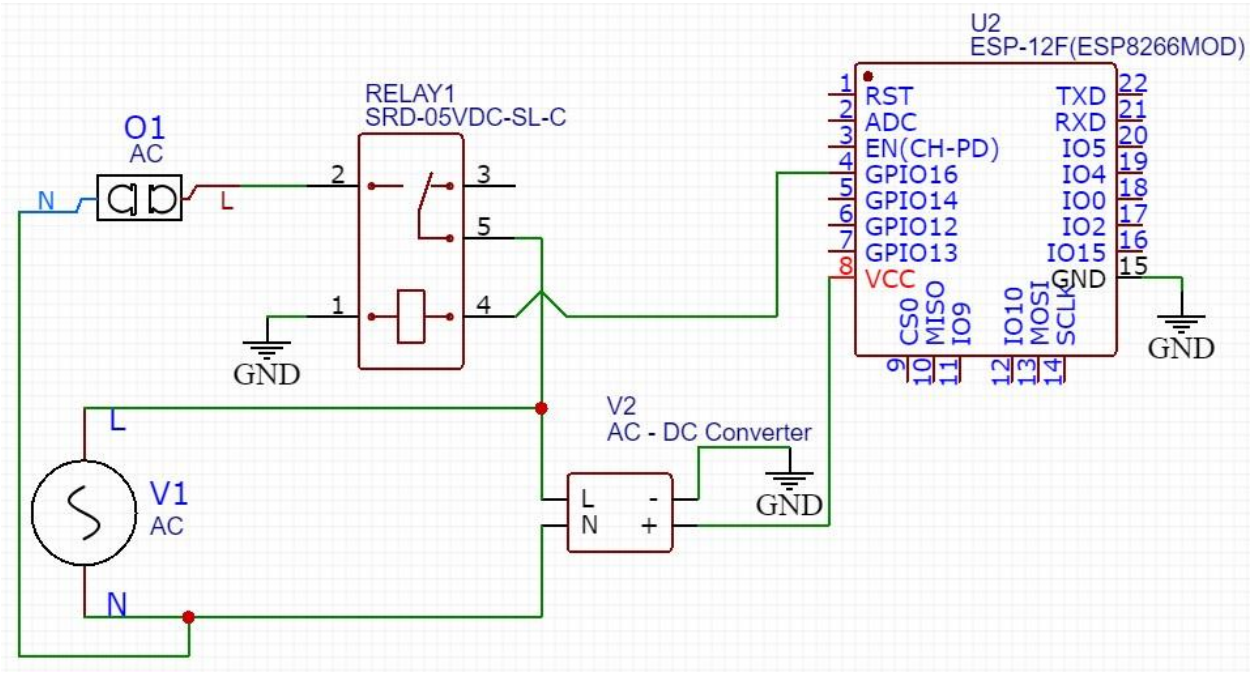


Рисунок 2.9 - Принципова схема прототипу розумної розетки

Цей пристрій, як і приведені вище прототипи датчиків, має дві стадії роботи. Перша стадія - ініціалізація аналогічна датчикам. Друга стадія функціонування пристрою відрізняється від такої у датчиків. Увесь час функціонування пристрій чекає команди від центрального пристрою у форматі JSON наступних видів: {"COMMAND": "ON"} чи {"COMMAND": "OFF"}, у відповідність з якою подасть і зніме напругу з навантаження

2.5 Мобільне застосування

На основі проведеного дослідження ринку домашньої автоматизації було виділено декілька основних способів взаємодії з системою. Серед яких звична фізична взаємодія з пристроями, управління за допомогою голосу. Проте найбільшу частину взаємодії в концепції системи «Розумного будинку» займає взаємодія за допомогою призначеного для користувача графічного інтерфейсу, який може бути представлений як у вигляді web- застосування, так і у вигляді мобільного застосування. Досліджувані представники ринку домашньої автоматизації віддають перевагу реалізації мобільного застосування. Цей вибір

грунтований на повсюдному поширенні смартфонів і планшетних комп'ютерів.

Розробка мобільного застосування завжди пов'язана з вибором платформи. Існує безліч різних операційних систем для мобільних пристроїв. Найбільшу популярність здобули Android від компанії Google і iOS від компанії Apple. За статистикою на квітень 2022 року, представленою на сервісі StatCounter [36], доля Android складає 75% від усього ринку, iOS покриває 22%. Зважаючи на обмеженість часу на розробку для реалізації мобільного застосування була вибрана платформа Android.

Для реалізації мобільного застосування під мобільну операційну систему Android в основному використовують одну з двох мов програмування : Java і Kotlin. На даний момент основна частина накопиченої інформації по розробки відноситься до Java, який був вибраний для реалізації мобільного застосування. Для розробки використовувалося середовище розробки Android Studio.

У концепції системи домашньої автоматизації мобільне застосування виконує дві основні функції: виступає дистанційним пультом управління, а також надає інформацію про стан системи. Виходячи з функціонального опису системи, що розробляється, необхідно реалізувати наступні компоненти: менеджмент пристроїв, менеджмент сценаріїв і чат з голосовим асистентом.

Мобільне застосування можна розбити на дві основні складові: графічний призначений для користувача інтерфейс і бізнес-логіку. Увесь досвід взаємодії з мобільним застосуванням грунтується на тому, наскільки правильно був спроектований інтерфейс. На цю тематику існує безліч різноманітних підходів до його проектування. Найбільш відомі концепції Якоба Нильсена «Десять евристик» [46], Бена Шнейдермана «Вісім золотих правила» [44]. Ці концепції націлені на те, щоб зробити взаємодії користувача з додатком максимально зручним і продуктивним.

Окрім представлених концепцій для розробки мобільного застосування для операційної системи Android існують рекомендації від компанії Google, які були описані в дизайн-системе Material Design 2.0.

На основі цих концепцій були розроблені дизайн мобільного застосування і схема основних екранів мобільного застосування (рисунок 2.10).

Схема основних екранів мобільного застосування включає 5 основних екранів, які демонструють графічний призначений для користувача інтерфейс основних компонентів системи : 1 — чат з голосовим асистентом, 2-3 — менеджмент пристроїв, 4-5 — менеджмент сценаріїв.

Детальніший опис екранів представлений в таблиці 2.

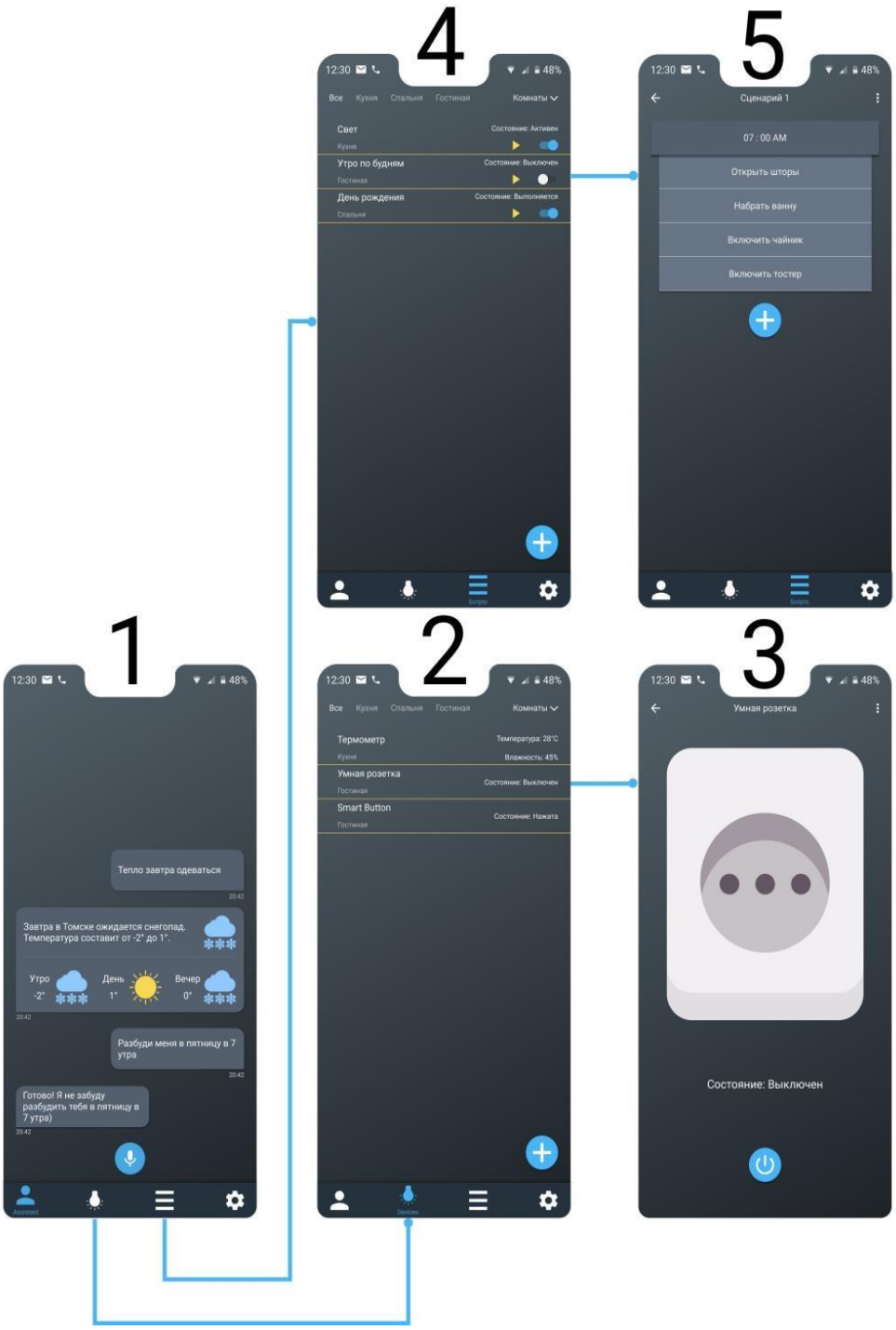
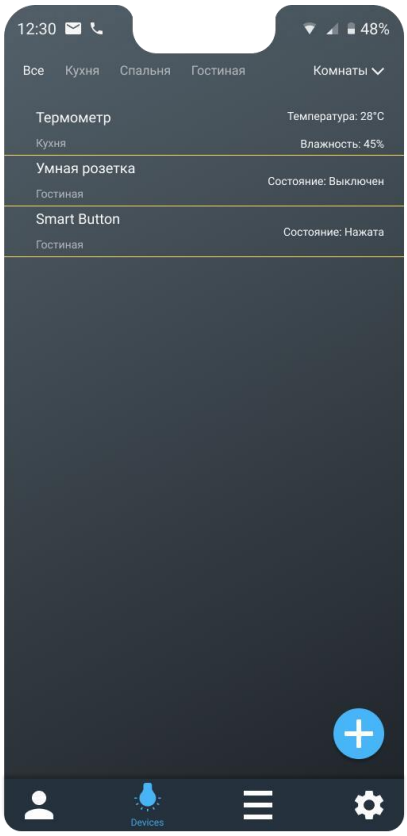


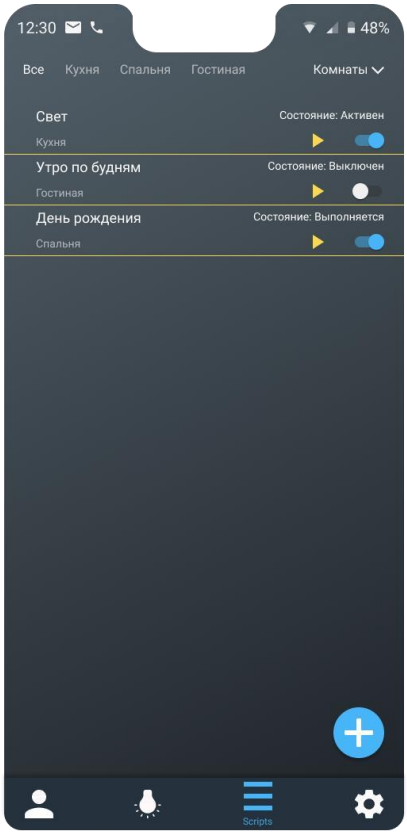
Рисунок 2.10 - Схема основних екранів мобільного застосування

Таблиця 2

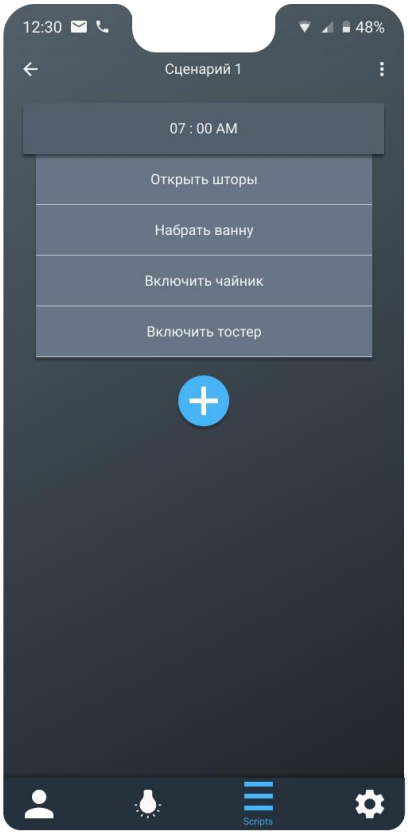
- Опис основних екранів мобільного застосування

Екран	Інтерфейс	Опис
1. Чат з голосовим асистентом		Цей екран реалізує взаємодію з голосовим асистентом. Увесь екран розбитий на три основні елементи: 1. Історію повідомлень 2. Кнопку початку прослуховування голосового запиту користувача 3. Меню
2. Список пристроїв		Екран зі списком пристроїв візуалізує коротку інформацію про усі пристрої в системі. Також з цього екрану додається новий пристрій, для цього необхідно натиснути кнопку +

Продовження таблиці 2

3. Взаємозв'язок з пристроєм		Екран взаємодії з конкретним пристроєм. Візуалізує зовнішній вигляд пристрою, текстову інформацію про стан пристрою і кнопки для управління пристроєм.
4. Список сценаріїв		Екран зі списком сценаріїв візуалізує коротку інформацію про усі створені сценарії. Також тут є присутніми кнопки для швидкої взаємодії з сценарієм: запуск сценарію, активація\деактивація відстежування сценарію.

Закінчення таблиці 2

5. Редактор сценарію		Екран створення\редагування сценарію. Представлений у вигляді тригера, списку дій і кнопки додавання нової дії. Для видалення\редагування вже доданої дії використовуються жести.

Висновки до розділу 2

У другому розділі було розроблено прототип інтелектуальної системи домашньої автоматизації «Розумний будинок». Визначено, що така система повинна будуватися на основі децентралізованої архітектури , що забезпечує дотримання принципів модульності та відмовостійкості, де кожен пристрій має власний мікропроцесор і пам'ять. Для практичної реалізації було обрано підхід створення мінімально життєздатного продукту (MVP). Даний функціонал включає організацію локальної бездротової мережі, менеджмент усіх компонентів, зберігання та обробку даних, виконання запитів користувача, а також взаємодію через мобільний додаток та голосові команди з підтримкою автоматизації алгоритмів на основі сценаріїв.

В якості центрального пристрою (хаба), який забезпечує об'єднання пристроїв в єдину мережу та їх менеджмент, було обрано одноплатний комп'ютер Raspberry Pi 3 b+. Він володіє достатньою продуктивністю, компактністю та має вбудовані модулі бездротового зв'язку. Програмне забезпечення для центрального пристрою розроблено мовою програмування

Python 3. Для вирішення проблеми розриву зв'язку під час ініціалізації нових пристроїв архітектуру хаба було доповнено окремим модулем WiFi на базі мікроконтролера ESP8266. Крім того, центральний пристрій успішно реалізує менеджер сценаріїв (на основі тригерів і послідовності подій) та забезпечує надійний зв'язок з мобільним додатком через активне (сокетне) та пасивне (REST) підключення.

Для реалізації кінцевих пристроїв типу Plug & Play, які здатні функціонувати як у зв'язці з хабом, так і автономно з мобільним додатком, базовим компонентом було обрано мікроконтролер ESP8266 з підтримкою стандарту бездротового зв'язку WiFi (IEEE 802.11 b/g/n). Програмне забезпечення для мікроконтролерів розроблялося в середовищі Arduino. На базі цього апаратного рішення було створено три діючих прототипи. Перший — це датчик температури і вологості, побудований на базі сенсора DHT11. Другий — датчик протікання води, в основі якого лежить Water Sensor для детектування порогового рівня рідини. Третім прототипом стала розумна розетка, що виступає виконуючим пристроєм для дистанційного керування живленням; вона реалізована за допомогою електромеханічного реле SRD - 05VDC - SL - C та додаткового AC-DC перетворювача для безпечної роботи зі змінною напругою.

Основним графічним інтерфейсом взаємодії користувача з системою виступає мобільний додаток, розроблений для операційної системи Android мовою Java у середовищі Android Studio. Графічний інтерфейс був спроектований з урахуванням ергономічних концепцій Якоба Нільсена, Бена Шнейдермана та рекомендацій Material Design 2.0 від компанії Google. Розроблений додаток складається з п'яти основних екранів, які реалізують увесь необхідний функціонал системи: чат з голосовим асистентом для обробки природної мови, списки для візуалізації інформації та управління підключеними пристроями, а також зручні інструменти для створення і редагування сценаріїв автоматизації.

РОЗДІЛ 3 РОЗРОБКА МОДУЛЯ ГОЛОСОВОГО КЕРУВАННЯ ТА ОБРОБКИ ПРИРОДНОЇ МОВИ ДЛЯ СИСТЕМИ «РОЗУМНИЙ БУДИНОК»

З популяризацією досліджень в області штучного інтелекту одного з головних завдань стала обробка природної мови, на даний момент вже є безліч робіт і підходів до вирішення цієї проблеми. Серед усіх завдань по обробці природної мови можна виділяють два основних напрям: завдання аналізу і завдання синтезу. Для взаємодії користувача з системою на природній мові вимагається вирішити завдання аналізу.

Завдання аналізу можна розбити на два великі етапи:

1. Розпізнавання мови і переклад її в текст.

2. Розпізнавання тексту з метою виявлення інформації.

На даний момент часу розпізнавання мови вже є безліч різних рішень по перекладу мови в текст. Найбільш популярні операційні системи за умовчанням оснащені подібними рішення. Їх можна зустріти як на дескотних операційних системах, таких як Windows і MacOS, так і на мобільних пристроях - Android і iOS.

Спочатку застосування таких рішень обмежувалося введенням тексту за допомогою голосу і не мало високої якості і затребуваності. На даний момент застосування перекладу мови в текст використовується великий в пошукових сервісах, наприклад, таке рішення використовують Google пошук і Яндекс пошук.

Значну популяризацію взаємодії з інформаційними технологіями за допомогою мови стало зростання використання голосових асистентів на мобільних пристроях. Для операційної системи Android таким став Google Assistant, у випадку з пристроями під управлінням iOS - Siri від компанії Apple.

Із зростанням популярності використання голосових асистентів, вони стали поширюватися на окремих пристроях - розумних колонок. Такі технологічні гіганти як Google, Apple, Amazon, Яндекс вже представили свої рішення.

Враховуючи таке широке використання голосової взаємодії з цифровими

технологіями, на ринку існує безліч платних і безкоштовних рішень по перекладу мови в текст .Обидва представлені рішення мають високу точність і мають схожі умови використання. Зважаючи на наявність за умовчанням Google Speech Recognition на пристроях під управлінням операційної системи Android вибір був зроблений на користь цього рішення.

Після перекладу мови в текст необхідно витягнути корисну інформацію з отриманого тексту. Для розробки модуля розпізнавання команди користувача, необхідно мати базу можливих запитів. Охопити увесь спектр можливих виразів на природній мові для носіїв української мови - украй складне завдання. Тому для розуміння можливих запитів користувача був зібраний набір запитів для голосових асистентів, доступних українською мовою. Такими асистентами були вибрані Siri від компанії Apple, Google Assistant від компанії Google і Алиса від компанії Яндекс. Результатом став набір з 730 унікальних запитів українською мовою в природній формі.

Для перевірки якості алгоритму розпізнавання семантики, що розробляється, отриманий набір був розбитий на тренувальну і тестову вибірки із співвідношенням 75% до 25% відповідно. Тренувальний набір використовувався для розробки алгоритму, для тестування використовувався тестовий набір. Такий підхід дозволяє оцінювати працездатність алгоритму на реальних нових даних, що дає реалістичніші дані про якість.

Аналіз тренувальної вибірки показав, що усі запити можна розділити на два основні класи:

1. Запити з параметрами. Це ті запити, в яких явно можна виділити об'єкт взаємодії, тип взаємодії і параметри, якщо вони є присутніми.

2. Запити загального вигляду. У таких запитах складно або неможливо виділити основні складові, які б дозволили судити про семантику запиту.

Обробку природної мови ведуть на різних мовах програмування. У випадках, коли важлива швидкість розпізнавання, наприклад, для серверів, де можливі тисячі запитів в секунду, можуть використати компільовані мови програмування, такі як C\C++ і Java. У випадку з низько навантаженими системами, такі як локальне розпізнавання в системі «Розумний будинок»,

можна використати скриптові мови. Які дозволяють швидко створювати прототипи і модифікувати їх. В якості такої мови був вибраний Python 3 і оболонка Jupyternotebook.

3.1 Розпізнавання запитів з параметрами

Виділення ключових складових запиту для витягання семантики — складне завдання, досліджуване з середини минулого століття. Велике застосування в цій проблемі знайшло створення парсера на контекстно-вільних граматиках. Контекстно-вільні граматики — це окремий випадок формальної граматики, у якої ліві частини усіх продукций являються поодинокими нетерминалами. Тобто до будь-якого нетерминалу можна застосувати продукцію незалежно від контексту цього нетермінала. Цей підхід дозволяє обробляти і виділяти структуру в тексті на основі правил [29].

Для запиту «Встанови будильник на 8 ранку в середу» явно можна виділити об'єкт взаємодії - «будильник», тип взаємодії - «встановити» і параметри: дата - «середовище» і час - «8 ранку». У простому випадку граматика прикмет наступний вид:

$$
\begin{aligned}
&S \rightarrow VERBNOUN\ PPPP \\
&VERB \rightarrow \text{'Встанови '} \\
&NOUN \rightarrow \text{'будильник'} \mid \text{ранок'} \mid \text{середа'} \\
&PP \rightarrow PREP\ NUMR\ NOUN \mid PREP\ NOUN \\
&PREP \rightarrow \text{'на'} \mid \text{'в'} \\
&NUMR \rightarrow \text{'8'}
\end{aligned}
\tag{3.1}
$$

За допомогою бібліотеки обробки природної мови Natural Language Toolkit(NLTK) було побудовано дерево розбору (додаток А) і розбір в текстовому виді (рисунок 3.1).

На основі цього запиту можна виділити об'єкт взаємодії, тип взаємодії і параметри. Проте слід зазначити, що ця граматика підходить виключно для цього запиту, допускаючи лише перестановку місцями параметрів. Але у разі

спрощення або ускладнення пропозиції ця граматика вже не спрацює, як і у разі, якщо слова стоятимуть в іншій формі (рід, відмінок, число і так далі). Для вирішення цієї проблеми необхідно описати всілякі словоформи і варіанти побудови пропозиції. Складання такої граматики вручну складне одноманітне завдання, яке можна спробувати автоматизувати.

```
(S[]
  (VERB[] Установи)
  (NOUN[] будильник)
  (PP[] (PREP[] на) (NUMR[] 8) (NOUN[] утра))
  (PP[] (PREP[] в) (NOUN[] среду)))
```

Рисунок 3.1 - Текстовий розбір запиту «Встанови будильник на 8 ранку в Середовище»

Це завдання можна розбити на дві підзадачі:

1. Динамічна генерація слів

2. Оптимізація граматики

Для того, що не вносити усі слова і їх форми, які можуть зустрітися в запиті, можна динамічно вносити їх в граматику під час розпізнавання запиту. Це можна зробити за допомогою морфологічних парсерів і POS- теггерів таких, як AOT, Mystem, NLTK POS, PyMorphy2. Зважаючи на доступність на Python 3, функціональному оснащенню, досить швидкості роботи і високої якості був вибраний Pymorphy2. Цей інструмент дозволяє визначати частину мови, рід, відмінок і інші атрибути слів українською мовою на основі словника OpenCorpa, а також здатний передбачати ці атрибути для слів, яких немає в словнику [32].

Оптимізацію складання граматики можна провести з використанням в граматиці замість слів - частини мови. Додатково можна об'єднати в групи іменні сутності з контекстом. В ході аналізу тренувальної вибірки були виділені наступні типи іменних сутностей: час, дата, подія, географічні об'єкти, додатки, валюта, персони, математика, вага, об'єм, довжина, номер телефону.

Реалізація вищеописаних підходів можлива за допомогою засобів NLTK, проте необхідно додатково створити деяку кількість різноманітних модулів по

первинній обробці запиту. Схожий необхідний функціонал спочатку має Yargy парсер [39]. Цей парсер призначений для обробки пропозицій українською мовою.

Результатом усіх описаних аспектів по складанню граматики і застосуванню її до запиту «Встанови будильник на 8 ранку в середу» за допомогою Yargy парсера являється наступне дерево розбору (рисунок 3.2) і текстовий варіант розбору (рисунок 3.3).

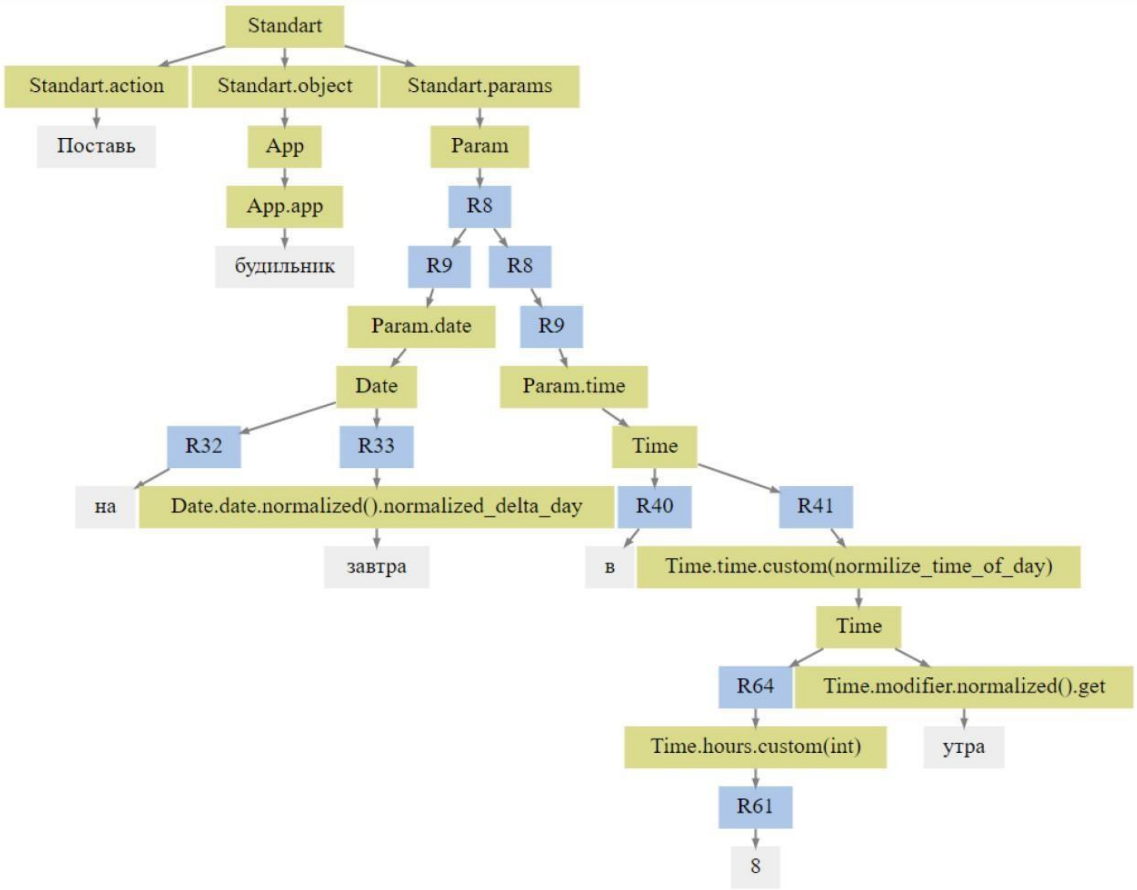


Рисунок 3.2 - Дерево розбору запиту «Встанови будильник на 8 ранку в середовище»

```
Standart(object=App(app='будильник'),
         action='Поставь',
         params=Param(currency=None,
                      date=Date(date=datetime.date(2019, 2, 26)),
                      event=None,
                      geox=None,
                      length=None,
                      person=None,
                      time=Time(time=datetime.time(8, 0)),
                      volume=None,
                      weigth=None,
                      value=None))
```

Рисунок 3.3 - Інтерпретація розбору запиту «Встанови будильник на 8

ранку в середу» у вигляді фактів

Використання цього підходу дозволяє створювати більше оптимізовані граматики, розпізнавати семантику 92% запитів. Серед усіх особливостей складання граматик, виділяючих Yargy парсер на тлі таких парсеров як Томита, NLTK, можна виділити основну - граматики складаються безпосередньо на мові програмування Python 3, що додає додаткову гнучкість до створення складних граматик. Також дерево розбору Yargy парсер дозволяє інтерпретувати у вигляді фактів, які також реалізуються на етапі складання граматик, що дозволяє легко надалі використати результати роботи парсера.

3.2 Розпізнавання запитів загального вигляду

У разі запитів загального вигляду складно або неможливо виділити об'єкт взаємодії і тип взаємодії в тому сенсі, який закладав користувач. Наприклад, на запит чи «Тепло мені сьогодні одягатися»? користувач чекає отримати відповідь про поточний стан погоди. Явно запит про відображення поточного стані погоди тут відсутній, тому використати контекстно-вільні граматики для виділення об'єкту взаємодії і типу взаємодії представляється неможливим.

Провівши аналіз підходів до вирішення цієї проблеми, можна виділити метод по створенню питально-відповідних систем. Їх головною особливістю є те, що в системі спочатку є присутньою база знань, яка містить кінцевий список запитів, для яких заздалегідь визначені відповіді. Система лише порівнює запит, що певним чином входить, з тим, що поступає в систему і відповідає за заздалегідь спланованим сценарієм.

Для розробки алгоритму по порівнянню запиту, що входить, з тим, що вже є у базі знань можна скористатися підрахунком відстані Левенштейна [3], тобто підраховувати кількості операцій, необхідних для перетворення одного слова на інше. Але таке порівняння ніяк не враховує семантичну різницю між словами, враховуючи лише морфологічні відмінності.

Іншим підходом за оцінкою схожості одного слова з іншим являється Word2Vec [16]. Ця технологія від компанії Google. Вона дозволяє відобразити

слово у багатовимірний векторний простір. Відображення слова здійснюється таким чином, що слова, які частіше зустрічаються в одному контексті, опиняються ближче в цьому просторі, а слова, які рідко або взагалі ніколи не зустрічаються в одному контексті, - далі. Цей метод не дає можливості упізнати системі значення слова, але дозволяє порівнювати слова на основі їх контексту.

Із застосуванням цього підходу можна порівнювати по формулі косинусної близькості (3.2) [16], як запит, що входить, схожий на ті, що є присутніми у базі знань. Далі необхідно лише вибрати найбільш схожий і виконати для нього заздалегідь визначений алгоритм. Варто відмітити, що в запиті також можуть бути присутніми деякі параметри, наприклад, в запиті чи «Тепло мені сьогодні одягатися»? у явному вигляді присутня дата. Для її витягання можна удатися до витягання іменованих сутностей на основі контекстно-вільних граматик.

$$Similarity = cos\,(\theta) = \frac{A*B}{\|A\|\|B\|} \qquad (3.2)$$

Для оцінки роботи алгоритму була складена база знань, що містить 63 запити, включаючи запит «Надівати сьогодні куртку»?, який повинен визначити алгоритм, як найбільш схожий для запиту, що входить, чи «Тепло мені сьогодні одягатися»?. У результаті роботи алгоритму отримуємо список відстаней між запитом, що входить, і усіма іншими (рисунок 3.4).

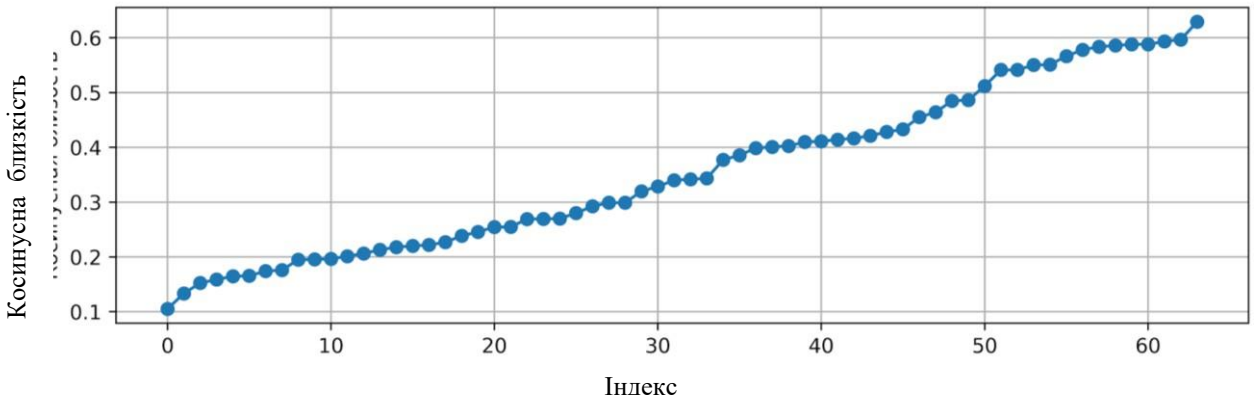


Рисунок 3.4 - Схожість запиту чи «Тепло мені сьогодні одягатися»?

Безліч значень функції косинусної близькості визначена від 0 до 1, таким

чином, що чим більше схожий запит, тим ближче значення буде до 0, чим менше схожий - ближче до 1. З рисунка можна відмітити, що найбільш близький варіант знаходиться на відстані 0.0897, що відповідає шуканому запиту. Подальша різниця у відстані між запитами зберігається упродовж усього видалення від запиту, що входить.

Подібний розподіл відповідає практично усім запитам, на яких робилася перевірка алгоритму, що говорить про працездатність алгоритму, але неможливості явного контролю в автоматичній режимі. Для точнішої ідентифікації шуканого запиту, а саме виділення його від інших слід провести первинну обробку запитів.

При вивченні роботи технології Word2Vec було відмічено, що одно слово в різній формі (рід, відмінок, число і так далі) є різними словами і вносять додаткову відстань. Для відвертання цього ефекту можна зводити слова до початкової форми, для цього підходить вищеописаний інструмент PyMorphy2. Додатково було помічено, що в деяких запитах трапляються слова, які не несуть семантичній значущості, так звані стоп-слова. До таких слів можна віднести: приводи, займенники, питальні слова і інші.

Застосувавши первинну обробку для запитів, що входять, і запитів, знаходяться у базі знань, отримаємо наступний список відстаней між запитом, що входить, і усіма запитами у базі знань (рисунок 3.5).

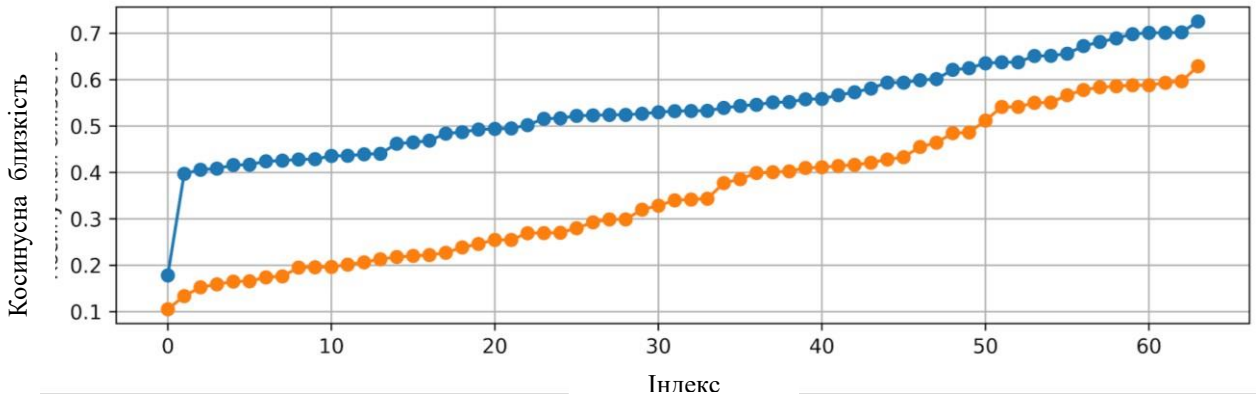


Рисунок 3.5 - Схожість запиту чи «Тепло мені сьогодні одягатися»? з первинною обробкою запитів

На рисунку 3.5 помаранчевим кольором представлено попередні розподіл відстаней між запитом, що входить, і запитами з бази знань, синім кольором відображений актуальний розподіл, отриманий з урахуванням первинної обробки запитів. Таким чином отримані результати значно відрізняються від первинних, вносячи різке збільшення відстані між найближчим запитом під індексом 0 і іншими, тим самим виділяючи його серед інших запитів. Це дозволяє в автоматичному режимі судити про правильність знаходження запиту.

Підсумком розробки алгоритму можна вважати високу якість пошуку алгоритму дії для запиту, що входить, шляхом порівняння його із запитами з бази знань. Цей алгоритм дозволяє порівнювати як запити, що трохи відрізняються, так і запити, складно побудовані з семантичної точки зору (таблиця 3).

Таблиця 3 - Приклади роботи алгоритму пошуку запиту у базі знань

Запит, що входить	Знайдений запит у базі знань
Розсміши мене	Розкажи анекдот
Про що ти мрієш?	У тебе є мрія?
Що сталося цього дня?	Розкажи про день в історії
Буде дощ?	Чи потрібно мені брати парасольку?
Мені нудно	Давай поговоримо

В результаті можна позитивно оцінювати цей підхід, як додатковий до розпізнавання за допомогою контекстно-вільних граматик. Що дозволяє збільшити якість роботи голосового асистента.

3.3 Укладення по розпізнаванню семантики

Розроблений алгоритм пошуку семантики забезпечує високу якість розпізнавання команди користувача. Алгоритм, грунтований на двох підходах: контекстно-вільних граматиках і порівняння схожості запитів, дозволив використати як структуровані, так і складні з семантичної точки зору запити.

Слід зазначити той факт, що використання контекстно-вільних граматик не вимагає від апаратної частини високопродуктивного рішення, що дозволяє розпізнавати такі запити безпосередньо на центральному пристрої. Використання ж навчених моделей для Word2Vec накладає певні вимоги. А саме, підсистема пам'яті, як оперативної, так і постійної, повинна мати певний мінімальний об'єм. В даному випадку потрібне 5 Гб пам'яті на постійному накопичувачі і 8 Гб оперативної пам'яті для моделі. Raspberry Pi 3 b+ володіє 1 ГБ оперативної пам'яті, що явно недостатньо для запуску модуля по порівнянню схожості запитів. Є два шляхи вирішення цієї проблеми:

1. Використати swap- файл в доповнення до оперативної пам'яті. Що дозволить використати локально порівняння запитів, що входять, із запитами з бази знань. Але ініціалізація займає тривалий проміжок часу, близько 20 хвилин.

2. Запускати REST- сервіс на видаленому сервері і обробляти запити по порівнянню на ньому. Такий підхід знімає необхідність у виборі іншого апаратного рішення для центрального пристрою і додаткового часу запуску. Проте необхідно реалізовувати видалений сервер.

Був вибраний другий підхід до вирішення цієї проблеми. Тим самим надалі забезпечивши незалежнішу зміну в цій компоненті системи. У разі оновлення моделі не треба буде вносити які-небудь зміни для центрального пристрою і оновлювати великий об'єм даних.

3.4 Укладення по голосовому асистентові

У цій роботі для вирішення підзадачі, пов'язаної з розробкою голосового асистента, розглядалася проблема аналізу запиту, що входить, з метою пошуку команди користувача. Повноцінний голосовий асистент також повинен мати зворотний зв'язок, тобто голосовою відповіддю на запит користувача. Для цього необхідно вирішити завдання синтезу:

Завдання синтезу складається з двох основних етапів:

1. Генерація відповіді в текстовому виді;

2. Генерація мови на основі тексту.

Зворотний зв'язок з користувачем можна підтримувати двома основними способами: заздалегідь підготовленими динамічними відповідями або використати генерацію відповіді за певними правилами.

Генерація відповідей - досить складне завдання, рішення якого не розглядається в цій роботі, оскільки вимагає глибокого занурення в тему і гідна окремого дослідження. Використання заздалегідь заготовлених відповідей не дозволить обхопити в належному об'ємі всілякі запити користувачів, проте такий підхід легкий в реалізації і достатній для створення прототипу системи домашньої автоматизації. Тому він був вибраний для зворотного зв'язку з користувачем.

Генерація мови на основі тексту складне завдання, яке не має закінченого рішення, але на ринку вже представлені рішення з високим показником якості. Ці рішення вже здатні генерувати мову схожу з людською, але що ще має явні відмінності. Для генерації мови використовувався вбудований за умовчанням в операційну систему Android модуль Text - To - Speech від компанії Google.

Висновки до розділу 3

У третьому розділі було розроблено спеціальну частину системи, яка відповідає за обробку природної мови та забезпечення голосової взаємодії користувача з системою «Розумний будинок». Для вирішення задачі перекладу голосу в текст було обрано технологію Google Speech Recognition, яка за замовчуванням доступна на пристроях з операційною системою Android. Зворотний зв'язок у вигляді синтезу голосової відповіді реалізовано за допомогою вбудованого модуля Text-To-Speech від Google на основі бази заздалегідь підготовлених відповідей. Щоб розробити і протестувати алгоритм, було зібрано набір із 730 унікальних запитів українською мовою, заснованих на командах до популярних асистентів, таких як Siri, Google Assistant та Аліса. Аналіз цієї вибірки дозволив розділити всі можливі запити на два основні класи: запити з параметрами та запити загального вигляду.

Для обробки запитів із параметрами застосовано підхід на основі контекстно-вільних граматик. Практична реалізація цього методу виконана за

допомогою Yargy парсера мовою Python 3. Крім того, для динамічного визначення атрибутів слів та зведення їх до початкової форми був використаний морфологічний аналізатор PyMorphy2. Такий комплексний підхід до структурування мови дозволив досягти якості розпізнавання семантики на рівні 92%, що є достатнім для виконання обчислень безпосередньо на центральному пристрої.

Що стосується розпізнавання запитів загального вигляду, де неможливо чітко виділити параметри, було розроблено питально-відповідну систему. Ця система базується на словнику з 63 еталонних запитів. Схожість запиту, що надходить від користувача, із базою знань обчислюється за формулою косинусної близькості з використанням векторного простору Word2Vec. Попередня обробка таких запитів, яка включає видалення стоп-слів та лематизацію, суттєво підвищила точність ідентифікації потрібного сценарію автоматизації.

У процесі реалізації системи виникли апаратні обмеження, пов'язані з високими вимогами навчених моделей. Зокрема, використання технології Word2Vec вимагає близько 8 Гб оперативної пам'яті, що значно перевищує можливості обраного центрального хаба Raspberry Pi 3 b+, який має лише 1 Гб. Для вирішення цієї проблеми розпізнавання складних запитів було винесено на віддалений сервер, з яким хаб взаємодіє через REST-сервіс. Це дозволило уникнути використання локального swap-файлу, ініціалізація якого займала б близько 20 хвилин, а також забезпечило додаткову гнучкість системи при майбутніх оновленнях мовної моделі без необхідності втручання в роботу центрального контролера.

РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

Охорона праці - система законодавчих актів, соціально-економічних, організаційних, технічних, гігієнічних і лікувально-профілактичних заходів та засобів, що забезпечують безпеку, збереження здоров'я і працездатності людини в процесі праці. Науково-технічний прогрес вніс серйозні зміни в умовах виробничої діяльності працівників розумової праці. Їх праця стала більш інтенсивним, напруженим, які вимагають значних витрат розумової, емоційної і фізичної енергії. Це зажадало комплексного рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці і відпочинку.

Охорона здоров'я трудящих, забезпечення безпеки умов праці, ліквідація професійних захворювань і виробничого травматизму складає одну з головних турбот людського суспільства. Звертається увага на необхідність широкого застосування прогресивних форм наукової організації праці, зведення до мінімуму ручної, малокваліфікованої праці, створення обстановки, що виключає професійні захворювання і виробничий травматизм.

Даний розділ магістерської роботи присвячений розгляду наступних питань:

- організація робочого місця програміста;
- визначення оптимальних умов праці програміста.

4.1 Опис робочого місця програміста

Робоче місце - це частина простору, в якому інженер здійснює трудову діяльність, і проводить велику частину робочого часу. Робоче місце, добре пристосоване до трудової діяльності інженера, правильно і доцільно організоване, у відношенні простору, форми, розміру забезпечує йому зручне положення при роботі і високу продуктивність праці при найменшому фізичному і психічному напрузі.

При правильній організації робочого місця продуктивність праці інженера

зростає з 8 до 20 відсотків.

Згідно ГОСТ 12.2.032-78 конструкція робочого місця і взаємне розташування всіх його елементів повинне відповідати антропометричним, фізичним і психологічним вимогам. Велике значення має також характер роботи. Зокрема, при організації робочого місця програміста повинні бути дотримані наступні основні умови:

• оптимальне розміщення устаткування, що входить до складу робочого місця;

• достатній робочий простір, що дозволяє здійснювати всі необхідні рухи і переміщення;

• необхідно природне і штучне освітлення для виконання поставлених завдань;

• рівень акустичного шуму не повинен перевищувати допустимого значення.

Головними елементами робочого місця програміста є письмовий стіл і крісло. Основним робочим положенням є положення сидячи.

Робоча поза сидячи викликає мінімальне стомлення програміста. Раціональне планування робочого місця передбачає чіткий порядок і сталість розміщення предметів, засобів праці і документації. Те, що потрібно для виконання робіт частіше, розташоване в зоні легкої досяжності робочого простору.

Моторне поле - простір робочого місця, в якому можуть здійснюватися рухові дії людини.

Максимальна зона досяжності рук - це частина моторного поля робочого місця, обмеженого дугами, що описуються максимально витягнутими руками при русі їх в плечовому суглобі.

Оптимальна зона - частина моторного поля робочого місця, обмеженого дугами, описуваними передпліччями при русі в ліктьових суглобах з опорою в точці ліктя і з відносно нерухомим плечем.

При проектуванні письмового столу варто враховувати наступне:

• висота столу повинна бути вибрана з урахуванням можливості сидіти

вільно, в зручній позі, при необхідності спираючись на підлокітники;

• нижня частина столу повинна бути сконструйована так, щоб програміст міг зручно сидіти, ні змушений підбирати ноги;

• поверхня стола повинна мати властивості, що виключають появу відблисків в полі зору програміста;

• конструкція столу повинна передбачати наявність висувних ящиків (не менше 3 для зберігання документації, лістингів, канцелярських приналежностей, особистих речей).

Важливим елементом робочого місця програміста є крісло. При проектуванні крісла виходять з того, що при будь-якому робочому положенні програміста його поза повинна бути фізіологічно правильно обгрунтованої, тобто положення частин тіла повинно бути оптимальним. Для задоволення вимог фізіології, що випливають з аналізу положення тіла людини в положенні сидячи, конструкція робочого сидіння повинна відповідати таким основним вимогам:

• Допускати можливість зміни положення тіла, тобто забезпечувати вільне переміщення корпуса і кінцівок тіла один щодо одного

• Допускати регулювання висоти в залежності від росту працюючої людини (у межах від 400 до 550 мм)

• Мати злегка увігнуту поверхню

• Мати невеликий нахил назад

Створення сприятливих умов праці і правильне естетичне оформлення робочих місць на виробництві має велике значення як для полегшення праці, так і для підвищення його привабливості, позитивно впливає на продуктивність праці. Забарвлення приміщень і меблів повинна сприяти створенню сприятливих умов для зорового сприйняття, гарного настрою. У службових приміщеннях, в яких виконується одноманітна розумова робота, що потребує значної нервової напруги і великого зосередження, фарбування повинна бути спокійних тонів - малонасичені відтінки холодного зеленого або блакитного кольорів.

При розробці оптимальних умов праці програміста необхідно враховувати освітленість, шум і мікроклімат.

4.2 Освітленість робочого столу

Раціональне освітлення робочого місця є одним з найважливіших факторів, що впливають на ефективність трудової діяльності людини, що попереджають травматизм і професійні захворювання. Правильно організоване освітлення створює сприятливі умови праці, підвищує працездатність і продуктивність праці. Освітлення на робочому місці програміста повинно бути таким, щоб працівник міг без напруги зору виконувати свою роботу. Стомлюваність органів зору залежить від ряду причин:

- недостатність освітленості;
- надмірна освітленість;
- неправильний напрямок світла.

Недостатність освітлення призводить до напруги зору, послаблює увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає осліплення, роздратування і різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку або профзахворювань, тому настільки важливий правильний розрахунок освітленості.

Штучне освітлення виконується за допомогою електричних джерел світла двох видів: ламп розжарювання і люмінесцентних ламп. Будемо використовувати люмінесцентні лампи, які в порівнянні з лампами розжарювання мають суттєві переваги:

- по спектрального складу світла вони близькі до денного, природного освітлення;
- володіють більш високим ККД (у 1.5-2 рази вище, ніж ККД ламп розжарювання);
- мають підвищену світловіддачу (в 3-4 рази вище, ніж у ламп

розжарювання);

• більш тривалий термін служби.

Розрахунок освітлення проводиться для кімнати площею 15 м^2, ширина якої 3.5 м, висота - 4 м. Скористаємося методом світлового потоку.

Для визначення кількості світильників визначимо світловий потік, падаючий на поверхню за формулою:

$$F = \frac{E \cdot K \cdot S \cdot Z}{n}$$, де

F - розраховується світловий потік, Лм;

E - нормована мінімальна освітленість, Лк (визначається за таблицею). Роботу програміста, відповідно до цієї таблиці, можна віднести до розряду точних робіт, отже, мінімальна освітленість буде E = 300 Лк при газорозрядних лампах; S - площа освітлюваного приміщення (у нашому випадку S = 15 м2); Z - відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1.1-1.2); К - коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників у процесі експлуатації (його значення визначається за таблицею коефіцієнтів запасу для різних приміщень і в нашому випадку К = 1.5); n - коефіцієнт використання світлового потоку (η = 1,1 ... 1.2).

4.3 Безпека в надзвичайних ситуаціях

Небезпека – це явища, процеси, об’єкти, інформація і самі люди, які можуть викликати небажані наслідки і призводити до погіршення стану здоров’я чи смерті людини, завдавати шкоди навколишньому середовищу й об’єктам господарської діяльності. Життєвий досвід людини показує, що шкоду людині може нанести будь-яка діяльність: робота на виробництві (трудова діяльність), різні види відпочинку, розваги та навіть діяльність, пов’язана з навчанням.

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

Небезпека об’єктивно існує в просторі і часі та реалізується у вигляді

потоків енергії, речовини, інформації.

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

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

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

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

Основними критеріями переходу небезпеки в надзвичайну ситуацію є:

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

Під час експлуатації ЕОМ, можуть мати місце наступні надзвичайні ситуацій: пожежі, аварії теплопроводу, водопроводу, ураження електричним струмом.

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

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

- помічено механічні пошкодження та інші дефекти електрообладнання і електропроводки;

- помітно підвищений рівень шумів при роботі на обладнанні;
- помітно підвищене тепловиділення від обладнання;
- постійно блимає екран монітора;
- стрибає текст на екрані монітора;
- відчувається запах горілого та диму;
- припинилася подача електроенергії.

Не приступати до роботи до повного усунення несправностей.

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

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

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

Висновки до розділу

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

ЗАГАЛЬНИЙ ВИСНОВОК

Комплексне дослідження предметної області систем домашньої автоматизації підтвердило, що цей високотехнологічний напрям перебуває у фазі активного експоненційного зростання. Він відкриває глибокі перспективи для подальших наукових розвідок та практичних інженерних розробок. Така позитивна динаміка обґрунтовується не лише стрімким збільшенням обсягів світового ринку та масштабними інвестиціями з боку глобальних технологічних корпорацій, а й стабільним зростанням зацікавленості серед кінцевих споживачів, які прагнуть оптимізувати та убезпечити свій житловий простір.

Водночас, як зазначають провідні аналітики галузі, на сьогоднішній день все ще відчувається певний дефіцит єдиного, стандартизованого уявлення про концепцію «Розумного будинку». Ця проблема стосується як виробників апаратного забезпечення, так і потенційних користувачів, що певною мірою гальмує процеси масової інтеграції таких систем у повсякденний побут. Однак, у міру природного еволюціонування ринку, програмно-апаратні комплекси від різних брендів починають набувати спільних концептуальних обрисів. Ця тенденція зумовлена глибоким аналізом користувацького досвіду та взаємним запозиченням найбільш вдалих інженерних рішень. Саме завдяки ідентифікації цих ключових спільних рис у межах даної кваліфікаційної роботи вдалося сформувати цілісне уявлення про сучасну модель інтелектуального житла.

Відповідно до актуальних тенденцій ринку IoT-технологій, повноцінна система повинна гарантувати користувачеві безперебійну можливість дистанційного керування, створення гнучких автоматизованих сценаріїв поведінки обладнання, а також підтримку природної взаємодії за допомогою голосових команд. Проведений структурний аналіз існуючих рішень довів, що фундаментальна архітектура таких комплексів незмінно базується на трьох основних стовпах: центральному контролері (хабі), наборі різноманітних кінцевих пристроїв (датчиках та актуаторах) і клієнтському мобільному додатку.

Спираючись на результати цього аналізу, було спроектовано узагальнену

архітектуру, яка стала надійним фундаментом для практичної розробки діючого прототипу системи «Розумний будинок». У цій структурі кожен елемент виконує суворо визначену функцію: центральний пристрій бере на себе роль головного координатора, що маршрутизує дані та управляє всіма процесами, тоді як периферійні модулі утворюють базовий шар екосистеми, забезпечуючи безпосередній контакт із фізичним середовищем, збираючи телеметрію та виконуючи фізичні дії.

Оскільки система домашньої автоматизації є надзвичайно складною та просторово розподіленою мережею, досвід взаємодії з нею концептуально відрізняється від використання традиційної побутової техніки чи стандартних інформаційних баз. Керування процесами розгортається одразу на трьох незалежних рівнях: класичному фізичному (через механічні або сенсорні вимикачі), дистанційному (через графічний інтерфейс смартфона чи планшета) та інтелектуальному (за допомогою голосових помічників). Ця багатовекторність є критично важливою особливістю, що виділяє розумні будинки на тлі інших технологій.

У підсумку проведеного дослідження було чітко сформульовано сучасний погляд на еволюцію домашньої автоматизації. Якщо на початкових етапах свого розвитку більшість систем обмежувалася лише базовим дистанційним увімкненням або вимкненням приладів, то з часом вони трансформувалися у високоінтелектуальні середовища. Цей прогрес супроводжувався впровадженням складних подієвих сценаріїв, предиктивної аналітики та нейромережевих асистентів, здатних розуміти контекст розмови.

Таким чином, головний вектор розвитку сучасних технологій остаточно змістився від звичайного віддаленого контролю до глобального фокусу на підвищенні якості життя людини. Цього вдається досягти завдяки глибокій автоматизації рутинних побутових процесів та безперервному моніторингу ключових показників. Подальша математична обробка зібраних масивів даних дозволяє системі приймати раціональні рішення, що не лише мінімізує витрати цінних ресурсів, а й гарантує безпрецедентний рівень захисту від аварійних ситуацій.

Звісно, на шляху масового впровадження цих інновацій все ще існують певні бар'єри, зумовлені відносною новизною та високою вартістю комплексних рішень. Не кожен споживач сьогодні готовий одразу інвестувати у повноцінну інфраструктуру, а купівля поодиноких смарт-пристроїв часто не розкриває всього потенціалу екосистеми, що може залишити хибне враження про справжню корисність технології. Тим не менш, стрімка позитивна динаміка зростання ринку, поступове здешевлення компонентної бази та потенційна підтримка ініціатив з енергоефективності дають усі підстави для максимально оптимістичних прогнозів. У найближчому майбутньому такі локальні розумні будинки неминуче інтегруються і стануть базовими структурними елементами глобальних масштабних проектів «розумних міст» (Smart City).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Аналіз ринку систем «Розумний будинок» в Украні [Електронний ресурс]. - URL: https://drgroup.ru/513-issledovanie-rinka-intellectualnih-domov.html (дата звернення : 30.03.2026).
2. Байгозин Д. В., Первухин Д. Н., Захарова Г. Б. Розробка принципів інтелектуального управління інженерним устаткуванням в системі «Розумний будинок» // Вісті політехнічного університету. Інжиніринг георесурсів. - 2008. - Т. 313. - №. 5.
3. Воску Л. С. Інтернет речей // "Нові інформаційні технології". Тези доповідей XX Міжнародній студентській конференції-школи-семінару. - 2012. - С. 89.
4. Галкін Д. В., Сербин В. А. Еволюція призначених для користувача інтерфейсів : від терміналу до доповненої реальності // Гуманітарна інформатика. - 2013. - №7.
5. Гаврилович Н. В., Сейтвелиева С. Н. Аналіз комерційних систем розпізнавання мови з відкритим API // Таврійський науковий оглядач. – 2016. – №. 6 (11).
6. Жданов А. А. Алгоритми побудови адаптивного мовного людино-машинного інтерфейсу для програмних систем: дис. - Фізико-технічний інститут, 2018.
7. Жирків А. Інтернет речей і хмарні технології Eurotech // Сучасні технології автоматизації. - 2015. - №. 2. - с. 6.
8. Китаев А. Е., Миронова И. И. Маркетингове дослідження ринку розумних будинків в цифровій економіці // International Journal of Open Information Technologies. - 2017. - Т. 5. - №. 10.
9. Куликів А. А. та ін. Побудова інформаційної системи «Розумний будинок» // Постулат. - 2018. - №. 4.
10. Кудусов І. І. «ІНТЕРНЕТ РЕЧЕЙ», «РОЗУМНИЙ БУДИНОК», «ІНТЕЛЕКТУАЛЬНЕ ВИРОБНИЦТВО» І КРЕАТИВНІ КОРИСТУВАЧІ // Електронний міждисциплінарний науковий журнал Інтернетнаука. - 2017- Т. 3. - №3. - с. 84-89.

11. Маклаев В. А., Соснин П. И. Питально-відповідне моделювання операційного простору в проектуванні сімейства автоматизованих систем // Автоматизація процесів управління. - 2014. - №. 2. - С. 86-97.
12. Намиот Д. Е., Шнепс-Шнеппе М. А. Пристрій для моніторингу активності в розумному будинку // International Journal of Open Information Technologies. - 2015. - Т.3. - №2.
13. Науменко А. М. та ін. Розробка питально-відповідної системи з нейромережевим навчанням на базі сучасних вільних технологій // Іннов : електронний науковий журнал. - 2017. - №. 2(31).
14. Провотар А. И., Клочко К. А. Особливості і проблеми віртуального спілкування за допомогою чат-ботів // Наукові праці Вінницького національного технічного університету. - 2013. - №. 3.
15. Сапин А. С., Большакова Е. И. Особливості побудови морфопроцессора української мови CrossMorhpy // Нові інформаційні технології в автоматизованих системах. - 2017. - №. 20.
16. Соловйов А. А., Пескова О. В. Побудова питально-відповідної системи для української мови: модуль аналізу питань // Нові інформаційні технології в автоматизованих системах. - 2010. - №. 13.
17. Філонов Д. Р. та ін. Питально-відповідна система для підтримки абітурієнтів з використанням сучасних мессенджеров // Моделювання і аналіз інформаційних систем. - 2018. - Т. 25. - №. 4. – С. 411-420.
18. Шелманов А. О. Дослідження методів автоматичного аналізу текстів і розробка інтегрованої системи семантико-синтаксического аналізу // дисс. канд. техн. наук. - 2015. - Т. 5. - с.17.
19. Шнепс-Шнеппе М. А. Як будувати розумне місто. Частина 1. Проект "Smart Cities and Communities" в Програмі ЄС Horizon 2020 // International Journal of Open Information Technologies. - 2016. - Т. 4. - №. 1.
20. Шовин В. А. Програма chatbot - чат-бот або віртуальний співрозмовник // Математичні структури і моделювання. - 2016. - №. 4 (40).
21. Adriano D. B. et al. Iot - based Integrated Home Security and Monitoring System // Journal of Physics : Conference Series. - IOP Publishing, 2018. - Т.1140. – №1. –

c. 012006.

22. Amri Y., Setiawan M. A. Improving Smart Home Concept with the Internet of Things Concept Using RaspberryPi and NodeMCU // IOP Conference Series : Materials Science and Engineering. - IOP Publishing, 2018. - T. 325. - №. 1. - C. 012021.

23. Ashton K. et al. That 'internet of things' thing // RFID journal. - 2009. - T. 22. - №. 7. - C. 97-114.

24. Hardeniya N. et al. Natural Language Processing : Python and NLTK. - Packt Publishing Ltd, 2016.

25. Kenter T., Borisov A., De Rijke M. Siamese cbow: Optimizing word embeddings for sentence representations // arXiv preprint arXiv: 1606.04640. - 2016.

26. Kodali R. K., Yerroju S. Energy Efficient Home Automation Using IoT // 2018 International Conference on Communication, Computing and Internet of Things(IC3IoT). - IEEE, 2018. - c. 151-154.

27. Korobov M. Morphological analyzer and generator for Ukrainian languages // International Conference on Analysis of Images, Social Networks and Texts. - Springer, Cham, 2015. - c. 320-332.

28. Le Q., Mikolov T. Distributed representations of sentences and documents // International conference on machine learning. - 2014. - C. 1188-1196.

29. Lobaccaro G., Carlucci S., Löfström E. A review of systems and technologies for smart homes and smart grids // Energies. - 2016. - T. 9. - №. 5. – C. 348.

30. Meng Y. et al. Securing Consumer IoT in the Smart Home : Architecture, Challenges, and Countermeasures // IEEE Wireless Communications. - 2018. - T. 25. – №. 6. – c. 53-59.

31. Mobile Operating System Market Share Worldwide [Електронний ресурс]. - URL: http://gs.statcounter.com/os-market-share/mobile/worldwide (дата звернення : 30.03.2026).

32. Nath R. K., Bajpai R., Thapliyal H. IoT based indoor location detection system for smart home environment // 2018 IEEE International Conference on Consumer Electronics(ICCE). - IEEE, 2018. - c. 1-3.

33. Patel V., Shah A. Android Based Smart Home Controlled System Using

Raspberry Pi // Journal of Control & Instrumentation. - 2018. - Т. 9. - №. 2. - С. 5-11.
34. Perevalov A. et al. Question Embeddings Based on Shannon EntropySolving intent classification task in goal - oriented dialogue system // Procedings of the 7th International Conference on Applied Innovations in IT. - 2019.
35. Reina D. G. et al. The role of ad hoc networks in the internet of things: A case scenario for smart environments // Internet of things and inter - cooperative computational technologies for collective intelligence. - Springer, Berlin, Heidelberg, 2013. - с. 89-113.
36. Rob Van Kranenburg : What is IoT? [Електронний ресурс]. - URL: https://www.theinternetofthings.eu/rob-van-kranenburg-what-iot (дата звернення : 30.03.2026).
37. Smart Speaker Shipments Reached a Record Breaking 7.5 Million Units in Europe in 4Q18 [Електронний ресурс]. - URL: https://www.idc.com/getdoc.jsp?containerId=prEUR145008619 (дата звернення : 30.03.2026).
38. Thapliyal H. et al. Amazon Echo Enabled IoT Home Security System for Smart Home Environment // 2018 IEEE International Symposium on Smart Electronic Systems(iSES) (Formerly iNiS). - IEEE, 2018. - с. 31-36.
39. The Eight Golden Rules of Interface Design Design [Електронний ресурс]. - URL: https://www.cs.umd.edu/users/ben/goldenrules.html (дата звернення : 30.03.2026).
40. Worldwide Smart Home Device Forecast, 2019-2023 [Електронний ресурс]. - URL: https://www.idc.com/getdoc.jsp?containerId=US44997319 (дата звернення : 30.03.2026).
41. Usability Heuristics for User Interface Design [Електронний ресурс]. - URL: https://www.nngroup.com/articles/ten-usability-heuristics/ (дата звернення : 30.03.2026).
42. 2018 Global Smart Home Market Forecast [Електронний ресурс]. - URL: https://www.strategyanalytics.com/access-services/devices/connected-home/smart-home/reports/report-detail/2018-global-smart-home-forecast (дата звернення: 30.03.2026).

43. Гуменний П. О., Мельник А. О. Архітектура та моделі функціонування систем розумного будинку на базі IoT-технологій // Вісник Національного університету «Львівська політехніка». Інформаційні системи та мережі. – 2020. – № 8. – с. 45-54.

44. Коваленко О. В., Сидоренко М. М. Забезпечення кібербезпеки в системах Smart Home: європейський досвід // Сучасний захист інформації. – 2021. – № 3. – с. 22-29.

45. Лисенко В. П., Марценюк В. П. Проектування бездротових сенсорних мереж для систем автоматизації будівель // Телекомунікаційні та інформаційні технології. – 2018. – № 2. – с. 67-74.

46. Borgeson K., Lund E. Energy efficiency in European smart homes: A review of IoT applications // Energy and Buildings. – Elsevier, 2020. – Vol. 211. – P. 109780.

47. Ткачук А. І., Романенко В. Д. Інтелектуальний аналіз даних в системах розумного будинку: алгоритми машинного навчання // Наукові вісті НТУУ "КПІ". – 2021. – № 4. – с. 34-42.

48. Rossi M., Bianchi G. Privacy and data protection in smart home environments under GDPR // Computer Law & Security Review. – Elsevier, 2019. – Vol. 35. – P. 105322.

49. Павленко О. В. Використання мікроконтролерів ESP8266 та ESP32 в бюджетних системах домашньої автоматизації // Електроніка та зв'язок. – 2019. – Т. 24. – № 1. – с. 55-61.

50. Захарченко І. В., Петренко О. М. Розробка голосових інтерфейсів для української мови у системах «Розумний будинок» // Штучний інтелект. – 2020. – № 1. – с. 78-85.

51. Fischer M., Weber A. Smart Home Platforms: Interoperability Challenges and Standardization in Europe // European Conference on Ambient Intelligence (AmI). – Springer, Cham, 2019. – P. 145-160.

52. Бойко В. І., Шевченко Т. Г. Хмарні технології в архітектурі децентралізованих систем домашньої автоматизації // Системи управління, навігації та зв’язку. – 2022. – Вип. 2 (68). – с. 90-95.

53. Грицюк Ю. І. Аналіз вразливостей протоколів бездротового зв'язку (ZigBee,

Z-Wave, BLE) у системах Smart Home // Захист інформації. – 2020. – Т. 22. – № 3. – с. 144-152.

54. Ковальчук О. М., Müller T. Застосування стандарту Matter для інтеграції гетерогенних пристроїв Інтернету речей // Оптикоелектронні інформаційно-енергетичні технології. – 2023. – Т. 45. – № 1. – с. 102-110.

55. Didenko O., Schmidt P. Edge computing applications in IoT-based smart homes: A European perspective // Journal of Cloud Computing. – Springer, 2021. – Vol. 10. – P. 1-15.