

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних технологій
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач відділення
комп'ютерних технологій
Наталія СТЕФУРАК / _____ /
підпис
« ____ » _____ 2024 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА
до кваліфікаційної роботи
освітньо-професійного ступеня «фаховий молодший бакалавр»
зі спеціальності 123 «Комп'ютерна інженерія»
на тему: «Автономна система розпізнавання голосових команд українською
мовою»

Студент групи КІ-41	Сергій КОДЬМАН	_____ (підпис)
Керівник роботи	Василь ПАВЛЮС	_____ (підпис)
Консультанти: з техніко-економічного обґрунтування	Любов МЕЛЕНЧУК	_____ (підпис)
нормоконтролер	Ольга СЛЄПЦОВА	_____ (підпис)

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних технологій
циклова комісія інформатики та комп'ютерних дисциплін

ЗАТВЕРДЖУЮ

Завідувач відділення

комп'ютерних технологій

Наталія СТЕФУРАК / _____ /
підпис

«__» _____ 2023 р.

ЗАВДАННЯ

на кваліфікаційну роботу

на здобуття освітньо-професійного ступеня «фаховий молодший бакалавр»

студенту Кодьману Сергію Володимировичу

(прізвище, ім'я та по-батькові студента)

1. Тема роботи: «Автономна система розпізнавання голосових команд українською мовою», затверджена наказом по коледжу від “27” листопада 2023 р., №234а-н.
2. Термін здачі студентом завершеної роботи “28” червня 2024 р.
3. Вихідні дані до проєкту: технічне завдання, наявні рішення на світовому ринку, сучасні моделі розпізнавання мовлення.
4. Перелік питань, які повинні бути розроблені в проєкті:
 - а) основна частина: аналіз предметної області, постановка завдання, проєктування системи, вибір компонентів, розробка алгоритму роботи, тестування системи.
 - б) техніко-економічне обґрунтування: аналіз ринку, розрахунок вартості проєктування, обґрунтування доцільності розробки.

5. Перелік графічного матеріалу: фото компонентів та їх будови, структурна схема мікрокомп'ютера, діаграми послідовності роботи та потоків даних в системі, блок схема алгоритму роботи, знімки результатів тестування системи.

6. Консультанти роботи:

Розділ	Консультанти	Підпис, дата	
		Завдання видано	Завдання прийнято
з техніко-економічного обґрунтування	Меленчук Л.І.		
	(вчена ступінь, звання		
	П.І.П. консультанта)		

КАЛЕНДАРНИЙ ПЛАН виконання кваліфікаційної роботи

№ п/п	Найменування етапу	Терміни	
		початку	завершення
1.	<i>Вибір теми, ознайомлення з вимогами до дипломного проєктування.</i>	20.11.23	27.11.23
2.	<i>Огляд типових рішень, технологій реалізації та написання відповідного розділу ПЗ.</i>	12.11.23	03.12.23
3.	<i>Розробка функціональних вимог до проєкту та робота над структурою системи. Написання відповідного розділу ПЗ.</i>	04.12.23	03.02.24
4.	<i>Проектування системи та написання відповідного розділу ПЗ.</i>	04.02.24	01.03.24
5.	<i>Реалізація та тестування системи. Написання відповідного розділу ПЗ.</i>	02.03.24	09.05.24
6.	<i>Опрацювання економічного розділу кваліфікаційної роботи.</i>	10.05.24	22.05.24
7.	<i>Робота над оформленням пояснювальної записки.</i>	23.05.24	17.06.24
8.	<i>Попередній захист кваліфікаційної роботи.</i>	18.06.24	18.06.24
9.	<i>Підготовка до захисту кваліфікаційної роботи.</i>	18.06.24	28.06.24
10.	<i>Захист кваліфікаційної роботи.</i>	28.06.24	28.06.24

Дата видачі завдання “27” листопада 2023 р. Керівник _____ / Василь ПАВЛЮС

Завдання прийняв до виконання _____ / Сергій КОДЬМАН

Реферат

Кваліфікаційна робота. Автономна система розпізнавання голосових команд українською мовою. 58с., 19 рисунків, 4 додатки, 8 джерел.

Об'єкт розробки – розпізнавання голосових команд, проєктування та розробка автономних систем зі штучним інтелектом на базі мікрокомп'ютера Raspberry PI та їх підключення до системи розумного будинку.

Метою роботи є розробка автономної системи розпізнавання голосових команд українською мовою та її з'єднання з системою розумного будинку для керування підключеними пристроями використовуючи голосові команди.

Систему потрібно реалізувати у вигляді фізичного пристрою, ядром якого буде мікрокомп'ютер, до якого підключений мікрофон, зі штучним інтелектом, який буде ядром для розпізнавання голосу та передачі відповідних команд до системи розумного будинку.

Найважливішими аспектами роботи цієї системи є відправлення точних команд та забезпечення коректної роботи системи з пристроями розумного будинку та алгоритмами їх роботи.

В процесі роботи над цим проєктом було використано велику кількість теоретичної інформації та практичних знань у сфері роботи з системами розумного будинку та їх компонентами і написанні алгоритмів роботи з використанням мови програмування Python.

Результатом виконання роботи є готова до використання система.

АВТОНОМНА СИСТЕМА, RASPBERRY PI, PYTHON, РОЗУМНИЙ БУДИНОК, ШІ, ГОЛОСОВІ КОМАНДИ, ШТУЧНА НЕЙРОННА МЕРЕЖА, ВІДПРАВЛЕННЯ КОМАНД.

Abstract

Graduation project. An autonomous command recognition system in Ukrainian. 58p., 19 figures, 3 appendices, 8 sources.

The object of development is voice command recognition, design and development of autonomous systems with artificial intelligence based on the Raspberry PI microcomputer and their connection to a smart home system.

The aim of the work is to develop an autonomous system for recognising voice commands in Ukrainian and connecting it to a smart home system to control connected devices using voice commands.

The system should be implemented in the form of a physical device, the core of which will be a microcomputer with a microphone connected to it, with artificial intelligence, which in turn will be the core for voice recognition and transmission of relevant commands to the smart home system.

The most important aspects of this system are sending precise commands and ensuring that the system works correctly with smart home devices and their algorithms.

In the process of working on this project, was used a large amount of theoretical information and practical knowledge in the field of working with smart home systems and their components and writing algorithms using the Python programming language.

The result is a ready-to-use system.

AUTONOMOUS SYSTEM, RASPBERRY PI, PYTHON, SMART HOME, AI, VOICE COMMANDS, ARTIFICIAL NEURAL NETWORK, SENDING COMMANDS.

ЗМІСТ

Скорочення і умовні позначки.....	6
Вступ.....	7
1 Аналіз існуючих рішень та постановка завдання	8
1.1 Аналіз предметної області.....	8
1.2 Огляд існуючих рішень	10
1.3 Постановка завдання.....	18
2 Проєктування системи.....	19
2.1 Аналіз компонентів	19
2.2 Обмін інформацією	25
2.3 Алгоритми роботи та структура системи	28
3 Реалізація та тестування системи	31
3.1 Вибір компонентів	31
3.2 Розробка програмної частини	37
3.3 Тестування системи	41
4 Техніко-економічне обґрунтування	45
4.1 Аналіз ринку	45
4.2 Розрахунок витрат на проєктування	46
4.3 Обґрунтування доцільності проєкту	48
Висновки	50
Перелік джерел посилання	51
Додатки.....	52

					КР.КІ 24.531.04.000 ПЗ		
Зм.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Кодьман С.В.			«Автономна система розпізнавання голосових команд українсько мовою»	Літ.	Арк.
Перевір.		Павлюс В.П.					5
Реценз.		Посвятовська О.Б.				ГФК.ВКТ.КІ-41	
Н. Контр.		Слепцова О.Я.					
Затверд.		Стефурак Н.А.					

СКОРОЧЕННЯ І УМОВНІ ПОЗНАКИ

ОС – Операційна Система

ПК – Персональний Комп'ютер

ПЗ – Програмне Забезпечення

ШІ – Штучний Інтелект

CPU – Central Processing Unit

IoT – Internet of Things

MQTT – Message Queue Telemetry Transport

OS – Operating System

RAM – Random Access Memory

ROM – Read Only Memory

SSH – Secure Shell

VNC – Virtual Network Computing

					КР.КІ 24.531.04.000 ПЗ	Арк.
						6
Зм.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Технології інтернету речей та ШІ з кожним роком стають все більш актуальними. Зокрема системи розумного будинку допомагають більш оптимально використовувати ресурси, наприклад воду або електроенергію, а дослідження ШІ проводяться стосовно найрізноманітніших сфер людського життя та життєдіяльності.

Сьогодні Україна є не останньою у дослідженні та впровадженні подібних технологій, наприклад:

- Такі провайдери як «Київстар», «Vodafone» та «Укртелеком» вже надають послуги зі сфери IoT, зокрема встановлення системи розумного будинку з різноманітними датчиками.

- Викладаються статті від українських дослідників у сфері ШІ, зокрема комп'ютерного зору або розпізнавання мови.

- Відомі українські заклади вищої освіти, і не тільки, створюють лабораторії для дослідження можливостей ШІ.

Якщо ж брати до уваги ситуацію у світі загалом, то за останні кілька років питання, що стосуються ШІ стали надзвичайно актуальними і це спровоковано стрімким ростом технологій і, відповідно, збільшенням його можливостей.

Що стосується IoT – його все частіше використовують у бізнесі для значного підвищення ефективності, а технології розумного будинку – це один з кращих прикладів впровадження користувацького IoT, який також набуває широкої популярності у наслідок кращої обізнаності людей.

Автономна система розпізнавання голосових команд українською мовою – це проєкт який об'єднує в собі сфери IoT, зокрема розумний будинок, та ШІ – розпізнавання мови і метою кваліфікаційної роботи є розробка саме такого прототипу. Основним завданням такої системи буде забезпечення користувачеві зручного керування за допомогою голосових команд та можливість її інтеграції до розумного будинку.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						7
Зм.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ПОСТАНОВКА ЗАВДАННЯ

Протягом цього та попереднього десятиліть активно велись дослідження, розробка та тестування різноманітних алгоритмів машинного навчання та створення моделей штучних нейронних мереж в різноманітних напрямках. Розпізнавання природньої мови хоч і не найпопулярніший напрям, проте і в рамках такого завдання вже існують готові рішення і проєкти на їх основі, а деякі вже роками знаходяться на ринку та користуються попитом.

1.1 Аналіз предметної області

Розробка автономної системи розпізнавання голосових команд українською мовою – це комплексний проєкт під час розробки якого слід дослідити предметну область.

Загалом, можна виділити кілька основних частин:

- Інтернет речей (IoT).
- Система розумного будинку.
- Штучний інтелект (ШІ).
- Фізичні компоненти.
- Визначення ризиків.

Почати варто з IoT, який є мережею фізичних пристроїв оснащених датчиками, програмним забезпеченням та можливостями підключення до мережі, які збирають та обмінюються даними.

Система розумного будинку, в свою чергу, є одним з видів користувацького IoT і передбачає його використання для автоматизації та керування різними аспектами будинку, такими як освітлення, опалення, вентиляція, кондиціонування повітря або безпека.

ШІ в системі розумного будинку може використовуватись для виконання наступних завдань:

					КР.КІ 24.531.04.000 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

– Аналіз отриманих даних означає використання певних алгоритмів, які дозволять системі вивчати поведінку користувача.

– Прогнозування є наслідком попереднього пункту і передбачає забезпечення певного рівня «самостійності» системи, наприклад – зміна температури на більш комфортну саме для користувача без необхідності дій з його боку.

– Розпізнавання голосових команд буде основним пунктом з частини ШІ в цьому проєкті та передбачає, що система розпізнавання голосових команд буде обробляти отримані звукові хвилі та на основі певного алгоритму надсилати сигнал для виконання певної функції розумного будинку.

Фізичні компоненти – це основа будь-якої системи розумного будинку, зокрема ними є датчики, актуатори та проводи, які забезпечують з'єднання між ними.

Зважаючи на комплексність цього проєкту існує чимало проблем з якими може стикнутись розробник на різних етапах виконання поставленого завдання:

– Безпека – це перший та найвагоміший ризик сучасних систем IoT будь-якого рівня. Викликана ця проблема великою кількістю водночас підключених компонентів та постійним обміном даних між ними, що значно ускладнює відстеження потоку даних та полегшує процес непомітного викрадення даних.

– Сумісність компонентів – це ще один з цілком можливих випадків несправності системи. Наскільки багато існує різноманітних версій датчиків, настільки ж і збільшується шанс того, що деякі з них не зможуть коректно працювати з іншими. Наприклад більшість датчиків від компанії «Xiaomi» будуть несумісними з датчиками від інших компаній, а їх інтеграція до власноруч спроектованої системи буде складним або, також, неможливим процесом.

Попри певні ризики та проблеми сучасних систем розумного будинку вони стають все більш актуальними і якщо брати для прикладу ринок IoT в Україні, то не зважаючи на нестабільну економіку за останні 2 роки, у 2023 році

					КР.КІ 24.531.04.000 ПЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

капіталізація ринку розумного дому становила близько \$50 млн, що в певній мірі вказує на актуальність цієї технології.

1.2 Огляд існуючих рішень

Розпізнавання голосових команд вже певний час активно використовується як основна частина функціоналу голосових асистентів, наприклад:

- Siri (Apple).
- Google Assistant (Google).
- Alexa (Amazon).
- Cortana (Microsoft).
- Bixby (Samsung).

В процесі дослідження найпопулярніших віртуальних асистентів з ШІ, які володіють функціями розпізнавання голосових команд та актуальних статей за минулий 2023 рік стало відомо наступне:

- В більшості топів, перші 3 місця посідають Google Assistant, Alexa та Siri. Також, в деяких з них згадується про Cortana та Bixby.
- Всі згадані асистенти легко інтегруються до систем розумного дому від відповідної компанії, наприклад Google Assistant та Google Nest.
- Деякі сучасні асистенти не підтримують розпізнавання української мови взагалі або цю функцію розвивають повільно і бібліотека з командами, які може розпізнати асистент є не великою.

Зважаючи на беззаперечну популярність та відомість Google Assistant, Alexa та Siri далі буде більш детально описаний їх функціонал.

Google Assistant – один з найвідоміших голосових асистентів сьогодення, функції якого не обмежені лише розпізнаванням мови, як і у двох наступних. Працювати з ним можна через смартфон, бо починаючи з виходу ОС Android 7.1 Google Assistant є частиною вбудованого у систему ПЗ.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

Також, він вбудований в більшість продуктів компанії та ОС на базі Android, наприклад:

- Wear OS – ОС для розумних годинників.
- Android TV – ОС для телевізорів.
- Android Auto – ОС яка використовується для автомобілів.

Як і всі сучасні віртуальні помічники – він здатен створити унікальну голосову модель власника, що може значно підвищити якість його роботи на пристрої протягом тривалого використання.

Основними перевагами Google Assistant є:

- Доступність – як згадувалось вище, цей віртуальний помічник є доступним майже на всіх пристроях з ОС Android, а на більшості сучасних смартфонів – вбудованим і його потрібно лише увімкнути.
- Інтеграція з продуктами від Google на високому рівні, в деякі він інтегрований, а з деякими може напряду працювати, наприклад YouTube, Google Map, Gmail або ж популярними додатками, наприклад – Spotify.

Основним недоліком цього асистента можна вважати збір даних про користувача без можливості відмови (інакше він не буде працювати) і це стосується більшості продуктів від Google.

Google Nest – це розумний динамік, який через Wi-Fi зв'язується з системою розумного будинку та дозволяє використовувати голосові команди для керування ним (рис. 1.1).



Рисунок 1.1 – Google Nest Mini

Для розпізнавання голосу в цій системі використовується Google Assistant. Серед переваг використання Google Nest, як ядра розумного будинку можна виділити наступне:

- Сумісність з великою кількістю сторонніх додатків та фізичних компонентів.
- Можливість створення сценаріїв роботи за допомогою додатку або голосового асистента.
- Повна підтримка пристроїв на базі ОС Android, які підключені до мережі.
- Можливість розпізнавати мову кількох користувачів без потреби зміни голосової моделі вручну.

Головним недоліком цього динаміка є відносно мала максимальна відстань на якій він може розпізнати голосову команду.

Siri – це віртуальний помічник з ШІ, розробку якого почали ще у 2007 році в незалежній від Apple компанії, але у 2011 році цю компанію, разом з правами на Siri викупили Apple і з цього моменту вони почали інтеграцію цього асистента у власні продукти.

Як продукт компанії Apple цей асистент є вбудованим на всіх пристроях, наприклад:

					КР.КІ 24.531.04.000 ПЗ	Арк.
						12
Зм.	Арк.	№ докум.	Підпис	Дата		

- iPhone.
- iPad.
- Apple Watch.
- Apple TV.
- HomePod.

Звісно, він має свої переваги серед інших віртуальних асистентів:

- Інтеграція з продуктами від Apple. Як і Google, компанія Apple володіє великою кількістю власних продуктів, в яких використовується виключно їх власний голосовий помічник, для прикладу Siri добре працює з Apple Music.
- Природна мова Siri на якісно вищому рівні в порівнянні з конкурентами, що робить його значно зручнішим для використання.

HomeKit Accessory Protocol (HAP) – це протокол для керування приладами підключеними до системи розумного дому. Цей протокол дозволяє керувати розумним будинком використовуючи девайси від Apple та додаток «Home», який доступний на пристроях з наступними ОС:

- iOS.
- iPadOS.
- watchOS.
- macOS.
- Apple TV.
- HomePod.

HomePod – це розумний динамік який дозволяє керувати компонентами системи розумного дому за допомогою HAP (рис. 1.2).

					КР.КІ 24.531.04.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13



Рисунок 1.2 – HomePod 2-го покоління

Програмним ядром цього динаміка є ШІ Siri, тобто він може приймати та опрацьовувати голосові команди користувача і відповідно до них виконувати певні дії, наприклад увімкнути телевізор.

Основними перевагами розумного динаміка від Apple є:

- Можливість роботи зі всіма пристроями компанії та більшістю додатків і сервісів.
- Потужний алгоритм шифрування даних, який є вбудованим у більшість продуктів компанії.
- Всі компоненти, які можна підключити до HomePod проходять ретельну сертифікацію.

– Якість звуку є найкращою серед конкурентів.

Певні недоліки в цієї системи також присутні:

- В значній мірі обмежений список можливих компонентів: розумний динамік від Apple підтримує близько 450, коли Google Nest – мінімум 10000.
- Невеликий набір функцій та сервісів відносно конкурентів.
- Ціна кратно вища ніж в конкурентів.

Amazon Alexa – це віртуальний помічник з ШІ розроблений однойменною компанією. Його було розроблено у листопаді 2014 року на основі синтезатора природньої мови – Ivona, який вони викупили в польської компанії у 2013 році.

Як і всі віртуальні помічники з цього списку Alexa інтегрована у більшість продуктів компанії-власника, наприклад:

- Fire TV.
- Всі додатки-магазини від Amazon.
- Amazon music.

Також, цей ШІ інтегровано в різноманітні продукти інших компаній:

- Деякі моделі автомобілів від Toyota та Lexus 2018 року випуску.
- Пристрої для розумного дому від таких компаній як Asus, LG або Wozart Technologies.
- Деякі ноутбуки та ПК від HP, Lenovo, Acer та Razer.
- Консоль Xbox від Microsoft.

Щодо переваг та недоліків цього віртуального помічника можна зазначити два основні пункти:

– Як основну перевагу слід зазначити те, що цей помічник здатен працювати з надзвичайно великою кількістю послуг від Amazon, від музики до онлайн-покупок.

– Значним недоліком для українського користувача буде процес його використання в зв'язку з тим, що більшість пристроїв з якими він працює є недоступними в Україні.

Одним з пристроїв в який вбудовано Alexa є Echo – серія компонентів з ШІ для розумного дому від компанії Amazon. Їх випуск почався в 2014 році, а розробкою займалися з 2011 року.

На сьогоднішній день існує чимало пристроїв цієї серії:

– Echo Dot – розумний динамік, найпопулярніше рішення для керування розумним домом серед розробок Amazon.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						15
Зм.	Арк.	№ докум.	Підпис	Дата		

– Echo Look – камера, яка поєднує в собі ШІ та алгоритми комп’ютерного зору.

– Echo Show – це розумний динамік з великим рідкокристалічним дисплеєм та фронтальною камерою.

– Echo Auto – це хост Alexa призначений для автомобілів, який підключається до телефону користувача через Bluetooth.

Як ядро для управління розумним домом від Amazon в Україні прийнято розглядати Echo Dot, бо цей пристрій з серії Echo є найбільш доступним і лише в нього є міжнародна версія (рис. 1.3).



Рисунок 1.3 – Echo Dot 5-го покоління

Україна, як і ряд інших країн не входить до списку тих, для яких є своя версія пристроїв з серії Echo.

Цей динамік має декілька значних переваг відносно конкурентів:

– Можливість підключення IoT пристроїв з використанням протоколу Zigbee, що може значно підвищити енергоефективність системи розумного дому та кратно підвищує кількість сумісних компонентів – близько 100000.

– Велика база доступних додатків та послуг від партнерів, наприклад Spotify або можливість викликати таксі, хоч і лише менша частина з них доступна в міжнародній версії.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						16
Зм.	Арк.	№ докум.	Підпис	Дата		

Основним недоліком його використання на території України є обмежена функціональність.

Кожен з вище перерахованих варіантів інтеграції віртуального помічника з функцією розпізнавання голосових команд є досить розповсюдженим в сучасному світі.

Якщо ділити 3 дослідженні варіанти по сферах використання, то можна зробити наступні висновки:

– Siri та розумний динамік HomePod є очевидним рішенням для користувача, який надає перевагу продуктам від компанії Apple. Цю систему можна охарактеризувати як надійне та добре захищене ядро розумного дому, тому вона підійде для того, кому потрібна надійність та безпека.

– Google Assistant та Google Home – це найдоступніше, на сьогоднішній день, рішення для централізації керування системою розумного дому, яке розробила, підтримує та розвиває загальновідома компанія. Досить широкий функціонал та велика кількість сумісних компонентів робить систему хорошим рішенням як для пересічного користувача, так і для експертів у сфері IoT.

– Alexa та серія розумних пристроїв для розумного будинку Echo надає найбільше можливостей як з фізичного, так і з програмного боку серед конкурентів в силу множини доступних компонентів, додатків і, відповідно, потенційного функціоналу системи розумного дому, тому ця система підійде для того, кому важливий широкий функціонал.

Кожен з досліджених варіантів буде хорошим рішенням для покращення взаємодії користувача та системи розумного дому, особливо зважаючи на те, що два з трьох розглянутих рішень вже підтримують українську мову, а Apple вже працює над тим, щоб український користувач міг керувати своїми пристроями використовуючи голосові команди рідною мовою.

1.3 Постановка завдання

Після ретельного дослідження сучасного ринку в сфері IoT, зокрема систем з розпізнаванням голосових команд які можна інтегрувати до власного розумного дому стало зрозуміло, що готові рішення вже є і найбільші IT-компанії світу підтримують їх активний розвиток.

Розробка автономної системи розпізнавання голосових команд українською мовою – це комплексне завдання яке потребує детального аналізу та постановки основних вимог на його основі для майбутньої системи.

Мінімальна робочий проєкт повинен бути фізично реалізованою системою яка складається з мікрокомп'ютера та мінімум одного мікрофона на його борту. Основною її функцією буде розпізнавання голосової команди після ключового слова/фрази або фізичної активації.

Також необхідно врахувати та реалізувати наступні пункти:

- Хороша якість звуку. Це ключовий момент для коректної роботи, тому необхідно реалізувати фільтрацію звуку апаратно або програмно.
- Передбачити можливі ризики з підключенням розроблювального прототипу до системи розумного дому. Тобто, з якими системами він працюватиме, а з якими можуть виникнути проблеми.
- Захист. Сучасні системи розумного дому є вразливими до кібератак, голосова модель користувача також є конфіденційною інформацією, тому необхідно забезпечити систему певними проколами захисту.
- Обмін даними між прототипом та системою розумного дому – це ще один поріг, бо передбачається, що він не матиме з'єднання з мережею Інтернет, отже доведеться реалізувати шлях для відправлення команд систему та отримання від неї необхідних даних.

Зважаючи на список поставлених вимог робота над кваліфікаційним проєктом є, в значній мірі, ретельною перевіркою знань студента.

2 ПРОЄКТУВАННЯ СИСТЕМИ

Після ретельного дослідження систем розпізнавання голосу на сучасному ринку слід скласти перелік компонентів, які є необхідними для робочого прототипу системи, схеми підключення та взаємодії між ними та дослідити сумісність компонентів.

2.1 Аналіз компонентів

Вибір компонентів системи – це основний етап перед безпосередньою реалізацією системи фізично, який значною мірою вплине на подальшу роботу над ним. Зокрема, основними аспектами цього дослідження є:

- Вибір оптимального рішення для виконання поставленого завдання або певного блоку функцій. Наприклад, для розпізнавання голосових команд точно використовуватиметься нейронна мережа.

- Дослідження та порівняння можливих апаратних та програмних рішень.

Зважаючи на специфікацію проєкту, розроблювальна система міститиме оптимальну кількість компонентів, що забезпечить її компактність.

Основними компонентами апаратної частини системи є:

- Мікрокомп'ютер буде головним компонентом системи. ШІ, зокрема моделі для розпізнавання голосу та голосових команд – це велика кількість інформації на основі якої система буде генерувати та відправляти команди до розумного будинку. Окрім збереження інформації, в апаратній частині необхідно буде її обробляти, а це означає високі вимоги до обчислювальних можливостей апаратної частини. Отже, одним мікроконтролером проблему не вирішити, тому для розробки цієї системи було обрано саме мікрокомп'ютер.

Основними перевагами таких машин зараз є співвідношення ціни та обчислювальних потужностей, наприклад деякі з них, не зважаючи на розмір, можуть зрівнятися з сучасними ПК, але ціна буде кардинально відрізнятись.

					КР.КІ 24.531.04.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		19

Щоб побачити, як різні компоненти взаємодіють між собою, на рисунку 2.1 наведено приклад архітектури найпростішого мікрокомп'ютера.

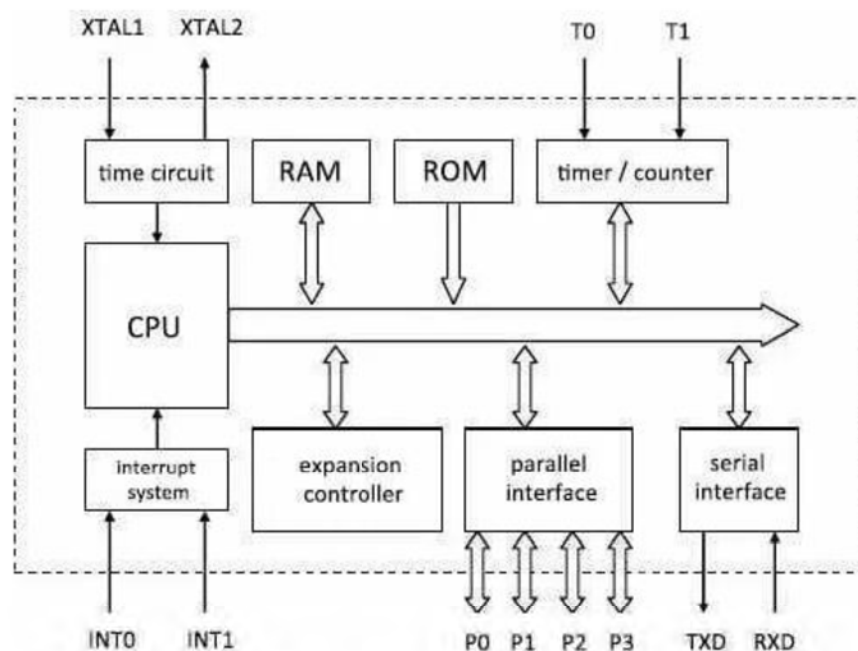


Рисунок 2.1 – Структурна схема мікрокомп'ютера

Як і в більшості видів обчислювальних машин, ядром мікрокомп'ютера є процесор. Головним аспектом, який відрізняє мікрокомп'ютер від мікроконтролера, окрім потужностей, є процес взаємодії користувача з машиною.

Для роботи мікроконтролера потрібно повністю прописати алгоритм його роботи певною мовою програмування, наприклад «C/C++», тоді як користувач взаємодіє з мікрокомп'ютером через оболонку ОС, графічного або консольного типу.

Отже, робота з мікрокомп'ютером знаходиться на значно вищому рівні в порівнянні з мікроконтролером.

Ще однією перевагою використання мікрокомп'ютера є методи взаємодії з ним, а саме:

- Фізично – шляхом підключення до нього монітора, чи іншого пристрою для виведення відео та додаткової периферії, наприклад клавіатура та комп'ютерна миша.

– Віддалено – використовуючи SSH або VNC-сервер. Наглядно процес взаємодії користувача з мікрокомп'ютером використовуючи ці протоколи показано на рисунку 2.2.



Рисунок 2.2 – Передача даних за допомогою SSH-протоколу

Аналогічно до рисунку, під час роботи з системою на місці сервера буде сам мікрокомп'ютер до якого можна підключитись з іншого ПК для подальшої роботи з ним, цей ПК і буде клієнтом, який показано на рисунку вище.

В обох випадках віддаленого підключення до мікрокомп'ютера використовується пара ключів: відкритий та закритий, відповідно для шифрування та дешифрування потоку інформації.

Різниця між цими протоколами полягає, здебільшого, в тому, що SSH-протокол передбачає взаємодію з системою виключно на консольному рівні, тоді як VNC – це повна візуалізація ОС мікрокомп'ютера.

– Мікрофон – це ще один невід'ємний компонент без якого розпізнавання голосу буде неможливим. Від нього залежатиме отримання вхідних даних для обробки.

Існує багато видів мікрофонів, але для проєкту потрібно розглянути наступні:

– Динамічний мікрофон (рис. 2.3).

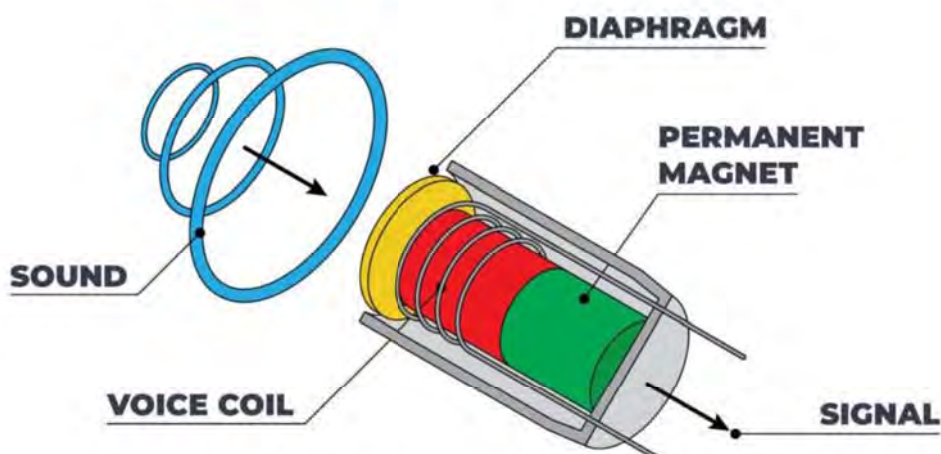


Рисунок 2.3 – Будова динамічного мікрофона

Мікрофони такого виду виділяються особливою надійністю та міцністю, що разом з малою вартістю робить їх одними з найпопулярніших, проте, зазвичай, вони підтримують вузький діапазон частот, що ускладнює підбір потрібного варіанту та роботу з ним.

За принципом роботи вони також прості, як показано на рисунку вище: звук змушує коливатись діафрагму, а вона в свою чергу рухає пружину навколо магніта, що створює електромагнітні коливання які перетворюються в аналоговий звуковий потік.

– Конденсаторний мікрофон (рис. 2.4).

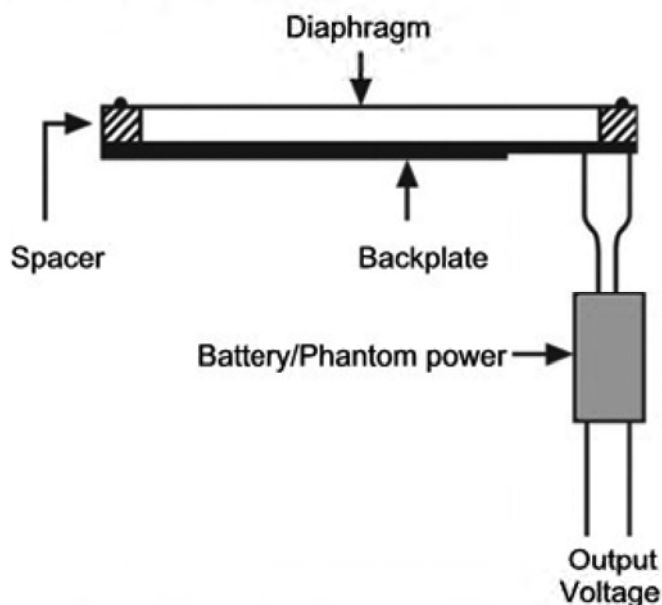


Рисунок 2.4 – Будова конденсаторного мікрофона

Цей вид мікрофонів виділяється широким діапазоном частот та чистим звуком, а основними недоліками є крихкість та вразливість до електромагнітних перешкод від інших електроприладів.

Кардинально відрізняється за принципом роботи від динамічного, як показано на рисунку вище, його принцип роботи наступний: коливання діафрагми генерують певний опір, а опір задньої стінки генерується на основі коливань повітря всередині, пізніше обидва сигнали проходять через посилювач сигналу (на рисунку – Phantom power (Фантомне живлення)) і на основі різниці опорів можна отримати звуковий потік.

Програмна частина проєкту, в основному, буде побудована на базі ШІ, а саме – готовій моделі розпізнавання природньої мови, навченій перетворювати аудіо-інформацію в текст.

Більшість сучасних моделей розпізнавання природньої мови є генеративними. Це означає, що для їх навчання використовують велику кількість датасетів – упорядкованих наборів даних, та певний генератор, який на основі готових даних створює відповідні фрагменти.

Використовуючи вхідні та згенеровані дані проводиться їх порівняння для визначення коефіцієнту точності, після порівняння перевіряється чи відповідають згенеровані дані поставленим умовам.

У випадку з розпізнаванням голосових команд, датасет буде складатись з двох масивів даних, де в першому буде аудіо-інформація, а в другому – текст, який відповідає їй. Для графічного представлення було створено загальну схему алгоритму машинного навчання генеративної моделі ШІ, яку показано на рисунку 2.5.



Рисунок 2.5 – Алгоритм навчання генеративного ШІ

Найпопулярнішими ресурсами, де можна безкоштовно отримати моделі розпізнавання природної мови є:

1) NeMo – проєкт від компанії NVIDIA який зосереджений на розпізнаванні Англійської мови.

2) wav2letter – один з багатьох напрямів дослідження генеративних моделей ШІ від компанії Meta, проте, як і з попереднім варіантом, українська мова – недоступна.

3) Deep speech – це інструмент для перетворення аудіо в текст від компанії Mozilla.

Головними перевагами цього рішення відносно попередніх є підтримка 30 різних мов світу, а для 10 з них доступні готові моделі розпізнавання мови і українська входить до цього списку.

Точність моделі для розпізнавання української мови сягає 30%, що є недостатнім в рамках проєкту, такий результат можна покращити шляхом навчання власноруч, проте це потребує багато ресурсів, даних та часу.

4) Alpha Cephei – це веб-ресурс, який пропонує автономні моделі розпізнавання мови які працюють за допомогою відкритого Vosk-API (vosk-моделі).

На момент дослідження ресурсу були доступні готові моделі для розпізнавання 18 різних мов світу включно з українською. На вибір пропонується 4 варіанти моделі розпізнавання української мови, крайньої на той момент, 3 версії:

– Nano – найменша з доступних, добре підходить для тестування роботи моделі. Здатна розпізнати близько 213 000 слів.

– Small – не зважаючи на назву є середньою за розміром на сьогоднішній день. Здатна розпізнати приблизно 1.7 мільйонів слів.

– Lgraph – модель, яку розробник позиціонує як динамічну, тобто за наявності достатніх знань її можна покращити.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						24
Зм.	Арк.	№ докум.	Підпис	Дата		

Найбільша доступна модель української мови налічує більш ніж 4.8 мільйонів слів, які вона може розпізнати та перетворити в текст.

Точність цих моделей розпізнавання української мови досягає 83%, що є хорошим результатом.

Тепер відомо з яких компонентів складатиметься система, тому для графічного представлення було створено структурну схему (рис. 2.6).

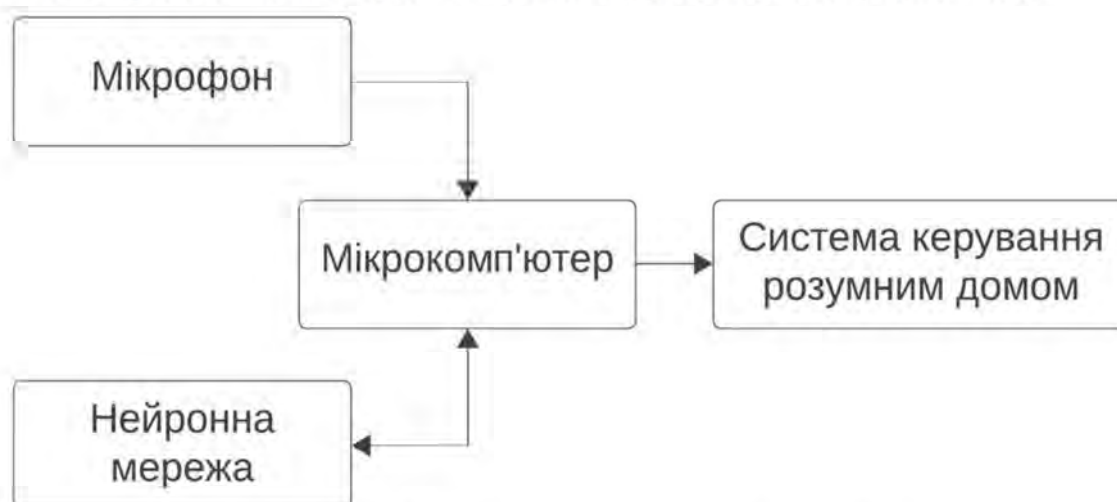


Рисунок 2.6 – Структурна схема системи

2.2 Обмін інформацією

Взаємодія між розроблювальною системою та розумним домом – це набір методів, проколів та інструментів, які дозволять реалізувати зв'язок між крайніми пристроями.

В процесі дослідження найпопулярніших рішень сьогодення було з'ясовано, що основними протоколами для обміну інформацією між системою розпізнавання голосових команд і розумним домом є Wi-Fi, Zigbee та Bluetooth.

Зважаючи на специфіку проєкту варто зазначити, що для реалізації зв'язку потрібно розглянути дротові та бездротові рішення.

Основним рішенням з дротової групи є USB-кабель, бо це означає надійний зв'язок, якому не завадять радіо-перешкоди. Також, сучасний USB-кабель дозволяє передавати великі об'єми інформації за короткий проміжок часу

та забезпечує надійне з'єднання між пристроями і ці аспекти можна вважати головними перевагами такого з'єднання.

Необхідність фізичного з'єднання, також може бути і вагомим недоліком, особливо у сфері IoT де ключовим фактором є те, що компоненти можуть знаходитись на певній відстані один від одного.

Беручи до уваги виключно ті, що дозволяють реалізувати передачу інформації між двома системами, можливими варіантами вирішення цієї проблеми можуть бути:

– Bluetooth – обмежений в радіусі з'єднання, але досить надійний варіант. В межах проєкту цей протокол може бути хорошим рішенням, бо основною вимогою до протоколу зв'язку є забезпечення обміну інформації між двома системами.

Бездротове з'єднання саме по собі є перевагою, але специфічним для конкретно цього рішення є досить мала відстань з'єднання – 10-15м без врахування можливих перешкод, що значно ускладнює роботу тих, хто захоче отримати доступ до даних.

Недоліками використання такого з'єднання може стати відносно мала відстань та швидкість передачі даних.

– Zigbee – це протокол низької споживчої потужності, який розроблено спеціально для передачі даних між компонентами розумного дому або значно масштабніших рішень IoT.

Передача невеликих об'ємів даних використовуючи zigbee буде надійною та ефективною, але передбачає необхідність зовнішніх або вбудованих модулів для його підтримки і якщо в системі розумного дому такий вже може бути, то для цього проєкту цей модуль стане додатковим невід'ємним компонентом. Враховуючи те, що на більшості мікрокомп'ютерів немає вбудованого Zigbee-модуля - це стане додатковими витратами, зокрема такий модуль вартує від 300 грн, що негативно вплине на можливість масового виробництва системи.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						26
Зм.	Арк.	№ докум.	Підпис	Дата		

З боку програмної реалізації передачі даних між кінцевими пристроями цей протокол можна виділити як складніший в порівнянні з альтернативами, це зумовлено специфікою його алгоритму роботи, а саме необхідність використання MQTT-брокера для обміну інформацією.

Якщо ж брати до уваги лише ті рішення, де буде використано цей протокол зв'язку, то він буде виділятися серед інших більшою надійністю та захищеністю, бо подібний сигнал дуже складно або неможливо перехопити без специфічних інструментів.

– Wi-Fi – загальновідомий та найпопулярніший серед відомих виробників розумних динаміків прокол передачі даних.

В більшості сучасних систем розумного дому по замовчуванню вбудований Wi-Fi-чип, як і в більшості мікрокомп'ютерів, включно з Raspberry Pi 4 model B, який використовуватиметься для реалізації системи розпізнавання голосових команд українською мовою.

Проте, з точки зору апаратної складової Bluetooth-модуль також зазвичай передбачений в пристрої, який використовуватиметься для системи розумного дому, тому було проведено дослідження роботи цих протоколів зв'язку і на основі отриманих даних можна провести їх порівняння виділивши 4 основні критерії:

– Дальність передачі даних двох згаданих проколів є кардинально різною, зокрема – специфіка роботи Bluetooth передбачена для комунікації між малою кількістю приладів на невеликій відстані, в той час коли Wi-Fi-сигнал може передаватися на десятки метрів.

– Енергоефективність Bluetooth-сигналу є високою, бо потребує малої кількості електроенергії в порівнянні з Wi-Fi, тому відповідні модулі на кінцевих пристроях прямо впливатимуть на це. В розроблювальній системі передбачається, що більшу частину часу роботи обидві сторони: система розпізнавання голосових команд та розумний дім, матимуть джерело постійного

живлення, а масштаб системи буде невеликим, тому критерій енергоефективності матиме невеликий вплив.

– Швидкість передачі даних дозволяє попередньо змодельовати ефективність системи в майбутньому. Серед порівнюваних проколів вища швидкість передачі у Wi-Fi, навіть останні версії Bluetooth не зрівняються з потенціалом першого, а передача даних – це основний пункт при виборі протоколу зв'язку для більшості сучасних розробок, які передбачають обмін інформацією.

– Безпека – це аспект, який значною мірою може відрізнитись від протоколу до протоколу. Обмежена відстань роботи Bluetooth є основним захистом цього протоколу, коли мережу Wi-Fi можна досить детально налаштувати шляхом встановлення Firewall для фільтрації трафіку.

2.3 Алгоритми роботи та структура системи

Вибір оптимальних алгоритмів роботи – це етап, на якому необхідно визначитись з тим, які алгоритми необхідно використовуватись для виконання ключових функцій системи.

Основні алгоритми, які буде використано у системи можна поділити на внутрішні та зовнішні.

До внутрішніх будуть відноситись ті, що необхідно реалізувати всередині ОС мікрокомп'ютера Raspberry Pi і далі буде наведено перелік таких алгоритмів:

– Ініціалізація мікрофона, який буде підключено окремо та можливість постійного і безперервного зчитування аудіо-потoku.

– Фільтрація вхідних даних. Для реалізації такого алгоритму необхідно використовувати вже готові програмні та апаратні рішення для зменшення фонового шуму.

– Перетворення отриманої аудіо-інформації в певний код або номер, який буде надіслано до розумного дому. Всередині основного скрипта необхідно передбачити функцію, яка задовільнить цю вимогу.

Після детально опису передбачених алгоритмів з'являється можливість їх графічного представлення, що допоможе в процесі реалізації. В якості графічного представлення було спроектовано блок-схему яку можна побачити на рисунку А.1.

Група зовнішніх алгоритмів включатиме лише реалізацію процесу передачі інформації між мікрокомп'ютером та розумним домом, як було описано вище.

Для узагальнення вище описаного, що стосується компонентів системи та передачі даних між ними було прийнято рішення скласти діаграму послідовності на якій буде показано в якому порядку система отримуватиме, опрацьовуватиме та передаватиме дані до розумного дому. Це дозволить графічно представити загальну структуру та краще зрозуміти те, як саме компоненти системи взаємодіятимуть між собою (рис. 2.7).



Рисунок 2.7 – Діаграма послідовності роботи системи

На схемі, наведеній вище, загально представлено вхідні та вихідні дані отримані за допомогою всіх передбачених алгоритмів, але для графічного представлення саме внутрішніх алгоритмів, від яких залежать основний функціонал проєкту, необхідно використати більш детальну діаграму – потоку

даних (англ. DFD). На такій схемі можна більш детально розглянути взаємодію між ключовими елементами розроблюваної системи (рис. 2.8).



Рисунок 2.8 – Діаграма потоків даних в системі

Проведення детального аналізу компонентів необхідних для коректної роботи системи допомогло визначитись з тими з них, які дозволять реалізувати функції передбачені на етапі планування, зокрема:

- Визначитись з ядром системи.
- Дослідити протоколи, які можна використати для передачі даних.
- Передбачити можливі ризики із застосуванням тих чи інших моделей системи.
- Визначитись з алгоритмами, які необхідно реалізувати для виконання запланованих функцій.

Наведені у цьому розділі схеми та алгоритми роботи системи допоможуть з майбутнім монтажем та написанням відповідних функцій.

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ

В попередніх розділах було проведено ретельні дослідження стосовно моделі майбутнього пристрою, в цьому ж буде детально описано процес розробки прототипу системи: визначення з конкретними компонентами, розробка алгоритмів роботи та тестування.

3.1 Вибір компонентів

Одним з висновків попереднього розділу було те, що для обчислювальних процесів необхідно використовувати потужний та компактний пристрій – мікрокомп'ютер.

Провівши аналіз доступних варіантів стало зрозумілим те, що існує мінімум 2 рішення:

– Raspberry Pi 4B – цю модель загальновідомого мікрокомп'ютера було розроблено ще у 2019 році, проте його потужність і, що не менш важливо, ціна є актуальним по сьогоднішній день (рис. 3.1).



Рисунок 3.1 – Raspberry Pi 4 model B.

Його характеристики та порти наступні:

– ARM-процесор з чотирма ядрами Cortex-A72 та тактовою частотою 1,8ГГц.

- 4Гб оперативної пам'яті типу LPDDR4 з тактовою частотою 3200Гц.
- Wi-Fi-модуль, також можливе підключення по Ethernet-кабелю.
- Bluetooth-модуль версії 5.
- 2 micro-HDMI та композитні порти для DCI дисплею та CSI камери.
- Порт для MicroSD яка використовується в якості основного накопичувача.
- 4 порти USB по 2 для версії 2.0 та 3.0.
- Живлення реалізовується від USB-C або GPIO напругою 5В та силою струму 3А в обох випадках.

Враховуючи те наскільки відомими є мікрокомп'ютери Raspberry Pi серед розробників, відповідно існує чимало аналогів, для прикладу – Banana Pi BPI-M5 (рис. 2.2).



Рисунок 3.2 – Banana Pi BPI-M5

Він є досить схожим на описаний раніше Raspberry Pi 4 model B і якщо розглядати виключно обчислювальні потужності, то в цьому рішенні вбудовано 4-х ядерний процесор архітектури Cortex-A55 з тактовою частотою 2ГГц, що означає його більшу швидкість роботи. Ще однією відмінністю між ними є ціна: вартість Raspberry Pi 4B - близько \$55, а Banana Pi зі схожими характеристиками,

а саме 4Гб оперативної пам'яті, – близько \$75, отже цей аналог майже на 50% дорожчий за обране рішення, що є одним з ключових аргументів в бік першого.

Зважаючи на співвідношення потужності та ціни, Raspberry Pi буде більш раціональним вибором. Також, важливо, що його розробники активно підтримують свої продукти і це значно зменшує можливі ризики виникнення проблем з ПЗ або подібним. Ці аспекти і стали основними для вибору цього мікрокомп'ютера в якості ядра системи.

Вибір мікрофона для системи важливий не менш ніж мікрокомп'ютер, бо необхідно врахувати наступний перелік вимог:

- Розмір, форма та тип підключення – це основні фізичні критерії, від яких потрібно відштовхувати в першу чергу.

- Чутливість повинна бути такою, щоб чітко передавати голос користувача і при цьому не записувати лишній шум.

- Діапазон частот більшості мікрофонів – від 20Гц до 20кГц, є допустимим, тому цей критерій найменш вагомий.

- Спрямованість – це також достатньо важливий критерій. Якщо розглядати розроблювальну систему з того варіанту, де мікрофон є безпосередньо на ній, то доцільніше буде використати широконаправлений, але як один з напрямів покращення системи можна використати масив одно-двонаправлених мікрофонів, що значно зменшить кількість шуму і, відповідно, підвищить якість її роботи.

Отже, будь-який мікрофон, що відповідає поставленим умовам можна вважати сумісним з системою. Це розширює можливості зі зміною або вдосконаленням та не несе за собою багато ризиків, бо Raspberry Pi, як і більшість UNIX-подібних систем підтримує велику кількість додаткової периферії.

Вагома різниця між різними мікрофонами також може полягати в типі підключення, зі всіх відомих слід розглянути 3 наступні:

					КР.КІ 24.531.04.000 ПЗ	Арк.
						33
Зм.	Арк.	№ докум.	Підпис	Дата		

– GPIO – передбачає пряме підключення до пінів, в цьому випадку – мікрокомп'ютера. Виділяється найнижчою вартістю та гіршою якістю звуку.

– Mini jack 3.5mm – підключається за допомогою відповідного інтерфейсу на платі, який наявний в моделях Raspberry Pi 4 та наступних. Головною перевагою є простота підключення та відсутність необхідності налаштувань.

– USB – надійний інтерфейс з високою швидкістю передачі даних. Найкраща якість звуку через стійкість до шуму, відповідно і ціна є вищою ніж у згаданих вище. Ризиком використання цього типу може бути відсутність ПЗ (драйверів) для ОС мікрокомп'ютера.

В кожного з описаних вище типів є перспективи для використання в проєкті, тому для візуального представлення того як їх можна підключити до обраного мікрокомп'ютера було спроектовано схему підключення, яку показано на рисунку 3.3.

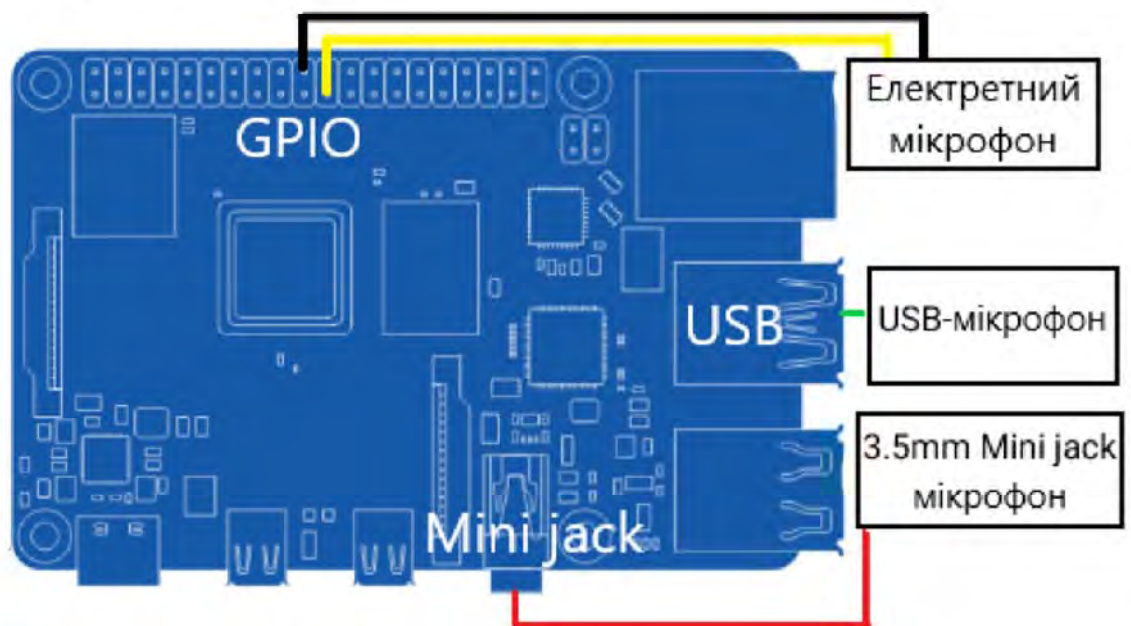


Рисунок 3.3 – Підключення мікрофонів

Огляд фізичних компонентів на цьому закінчується, а у висновок варто зазначити, що мала кількість компонентів означає легке масштабування через малі ризики з боку сумісності компонентів.

Окрім фізичних компонентів, слід визначитись з інструментами для розробки програмної частини, тому наступним пунктом буде вибір мови програмування за допомогою якої буде побудована логіка всього проєкту.

Використання мікрокомп'ютера означає, що вибір необмежений, проте більшу частину роботи буде покладено на модель нейронної мережі, а сьогодні більшість алгоритмів зі ШІ розроблено з використанням мови програмування Python яка вже не перший рік є однією з найпопулярніших у світі та активно підтримується як з боку розробників, так і з боку користувачів, які є авторами великої кількості зовнішніх бібліотек для виконання найрізноманітніших завдань.

У попередньому розділі згадувалось про Deep speech та Alpha Cerpei, як веб-ресурси де можна безкоштовно отримати готові моделі розпізнавання природної мови, зокрема і української.

Номинальна точність моделей кардинально відрізняється – розробники Deep speech пишуть, що точність їх моделі розпізнавання не перевищує 30%, а в Alpha Cerpei заявляють, що точність їх моделей приблизно дорівнює 80%.

Точність – це не єдина перевага моделей від Alpha Cerpei, якщо для використання їх моделей необхідно лише завантажити та вказати на них в скрипті, то моделі Deep Speech потребують детальних налаштувань перед використанням.

Саме ці два аспекти і те, що доступних моделей для розпізнавання української досить мало, стали вагомими причинами для зосередження на використанні саме vosk-моделей від Alpha Cerpei.

Ці моделі є потужним інструментом для розпізнавання природної мови і мають декілька основних складових, саме вони впливають на потенційну точність:

– Фонетичний словник – це перший та, ймовірно, найважливіший шар таких моделей, який розбиває слова на фонемі, які будуть основними

складовими в процесі передбачення слів. Відповідно він є ще й сховищем таких фонем, тобто звуків.

– Акустична модель є прямим продовженням фонетичного словника і, якщо перший – це звуки, то акустична модель – це процес статистичного передбачення того яке слово може бути утворено з отриманої послідовності звуків.

– Модель мови – це, в основному, підтвердження передбачень акустичної моделі, тобто на цьому етапі перевіряється статистична ймовірність того, що запропоноване нею слово може знаходитись на цьому місці, також враховуючи відповідність граматичним правилам мови і сенсу, який несе це слово.

Модульність побудови таких моделей розпізнавання мови є найефективнішим на момент роботи над проєктом, тому це ще один аргумент щодо використання vosk-моделей.

В попередньому розділі було детально розглянуто протоколи зв'язку, які доцільно використовувати в рамках проєкту, а саме: Bluetooth, Wi-Fi та Zigbee. Всі вони, окрім Bluetooth, використовуються тими чи іншими великими компаніями для власних моделей розумних динаміків і це ще раз доводить, що використовувати потрібно один з них.

Зважаючи на ймовірність того, що розумний дім може складатись з багатьох вузлів, кожен з яких приймає та відправляє дані стає зрозумілим, що налаштування великої системи стає надзвичайно складним, якщо розглядати Wi-Fi як основний зв'язок між ними. Саме тому для зв'язку між ними використовують більш «легкий», спеціалізований прокол – Zigbee, за умови, що компонент за замовчуванням не працює через Wi-Fi.

Проте, в рамках поставленого завдання метою є передача даних лише між двома вузлами: мікрокомп'ютером та ядром розумного дому, яким зазвичай є мікрокомп'ютер або мікроконтролер, тому раціональніше буде використати метод передачі HTTP-запитів через локальну мережу. Звісно, це може означати потребу додаткових налаштувань самого ядра розумного дому, але у випадку

					КР.КІ 24.531.04.000 ПЗ	Арк.
						36
Зм.	Арк.	№ докум.	Підпис	Дата		

використання Zigbee-протоколу, окрім налаштувань знадобляться додаткові модулі як для системи розпізнавання голосових команд так і для ядра.

Отже, передача команд за допомогою локальної мережі буде кращим вибором в рамках поставленого завдання з програмної та апаратної точок зору, також, це позитивно вплине на собівартість.

3.2 Розробка програмної частини

Початком роботи з більшістю сучасного ПЗ можна вважати вибір бібліотек, які необхідно використати для реалізації запланованого функціоналу. В процесі проєктування, було визначено перелік бібліотек які допоможуть виконати поставлене завдання:

- PyAudio представляє собою великий набір класів, методів та функцій які значно спрощують роботу зі звуком.

- Numpy є однією з найпотужніших та популярних бібліотек, методи якої зосереджені на оптимізації роботи з математичними функціями та роботі з масивами даних різних типів та розмірів, зважаючи на те, що більшість інформації можна логічно представити масивом даних – це робить її використання найбільш актуальним.

- Vosk – це набір інструментів для створення, навчання та використання моделей на основі Kaldi, що є основою для багатьох сучасних моделей нейронних мереж, зокрема для моделей від компанії Alpha Cephei, одну з яких і буде використано в процесі розробки.

- SciPy буде використана для фільтрації звукового сигналу всередині програми.

- Flask буде доцільно використати для створення сервера на мікроконтролері для роботи з HTTP-пакетами, за допомогою яких буде реалізовано процес передачі даних.

– Flask-SocketIO є доповненням до основної бібліотеки і дозволяє реалізувати більше налаштувань для роботи сервера, наприклад чергу відправлення команд.

Всі описані вище інструменти будуть необхідними для реалізації запланованого функціоналу.

Першим кроком в роботі скрипта буде ініціалізація мікрофона з певними налаштуваннями:

```
mic = pyaudio.PyAudio()
stream = mic.open(format=pyaudio.paInt16, channels=1,
rate=SAMPLE_RATE, input=True, frames_per_buffer=CHUNK_SIZE)
stream.start_stream()
```

Аргументи в методі «open()» вказують на те, що аудіо-інформація отримана з мікрофона буде змінена так, щоб для подальшої роботи використовувався цифровий потік з одним каналом, що буде критично важливим в подальшій роботі зі звуком.

Як згадувалось на етапі проєктування – фільтрація звуку, апаратна або цифрова, може значно покращити роботу моделі розпізнавання природної мови, тому було прийнято рішення використати функцію, яка буде видаляти частини масиву в залежності від заданих аргументів:

```
def butter_lowpass(cutoff, fs, order=5):
    nyq = 0.5 * fs
    normal_cutoff = cutoff / nyq
    b, a = butter(order, normal_cutoff, btype='low', analog=False)
    return b, a
```

Ось так, за допомогою функції «butter()» було визначено ймовірні межі за якими починається шум і на їх основі можна фільтрувати інформацію, і це використовується в наступній функції, яка використовує отримані дані:

```
def lowpass_filter(data, cutoff, fs, order=5):
    b, a = butter_lowpass(cutoff, fs, order=order)
    y = lfilter(b, a, data)
    return y
```

На цьому етапі обробка звуку є завершеною, тому отримані дані тепер придатні для обробки моделлю розпізнавання мови. Робота з нею, як і у випадку з мікрофоном, починається з ініціалізації:

```
model = vosk.Model("model")
recognizer = vosk.KaldiRecognizer(model, SAMPLE_RATE)
```

Доступні моделі для розпізнавання української мови є невеликими відносно, наприклад, англійської (150Мб – українська та 3Гб - англійська), проте, не зважаючи на розмір, найбільша з них навчена розпізнавати близько 5 000 000 слів. Саме цю модель з назвою «vosk-uk-large» було завантажено та додано до файлів проєкту.

Одним з ключових моментів роботи коду буде функція, яку показано в лістингу 3.1.

Лістинг 3.1 – Отримання, фільтрація та обробка звуку

```
def micro_read():
    data = stream.read(CHUNK_SIZE)
    np_data = np.frombuffer(data, dtype=np.int16)
    # Застосування фільтра до аудіо даних
    filtered_data = lowpass_filter(np_data, cutoff=3000,
    fs=SAMPLE_RATE, order=6)
    # Перетворення відфільтрованих даних назад до байтів
    filtered_data_bytes = filtered_data.astype(np.int16).tobytes()
    if recognizer.AcceptWaveform(filtered_data_bytes):
        return recognizer.Result()[14:-3]
```

Її важливість полягає у тому, що на цьому етапі отримується результат розпізнавання мови. Відповідно до того, як послідовно розділені стрічки коду можна описати алгоритм його роботи:

– Спершу виконується запис звукової доріжки розміру який задано в константі «CHUNK_SIZE» і виконується його перетворення з байтів у певні числа типу даних «int16», який відповідає за цілі числа розміром до 16 байтів.

– За допомогою функції «lowpass_filter()», яка описувалась вище, отриманий масив чисел фільтрується.

– Останнім кроком є повернення відфільтрованого масиву чисел назад у байти і спроба обробки записаного звуку моделлю розпізнавання природної мови, якщо було отримано результат, то він повертає лише текст за допомогою використання зрізу масиву.

Результатом розпізнавання звуку є масив, чимось схожий на файл типу «.json», тобто інформація подається у такому вигляді: «{ “увімкни світло” }», саме тому з масиву дістаються дані в діапазоні від 14 по -3 елементи.

На основі отриманих даних виконується формування остаточної команди у функції, яку показано в лістингу 1 додатку Б. Алгоритм, який використовується в цьому прикладі є лінійним та досить простим, але при цьому надає можливість оперативної зміни набору можливих команд, тобто будь-яку його частину можна редагувати або видалити, а додавання інших можливих варіантів, також, не займе багато часу.

Саме тому, для тестового варіанту було використано мінімальний набір можливих варіантів, а форма подачі у вигляді стрічки з 2 чисел дозволить зменшити навантаження на ядро розумного дому, особливо, якщо ним є мікроконтролер, який не є особливо потужним.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						40
Зм.	Арк.	№ докум.	Підпис	Дата		

Заключним етапом циклу роботи програми буде додавання сформованої команди до черги:

```
def listen_for_commands():  
    while True:  
        command = micro_read()  
        if command:  
            command_queue.put(command)  
            time.sleep(0.1)
```

Та відправлення її у відповідь на запит з боку користувача:

```
def handle_commands():  
    while True:  
        command = command_queue.get()  
        to_send = process_voice_command(command)  
        socketio.emit('response', {'command': to_send})
```

Отже, команда повинна відправлятися сервером клієнту, який до нього підключений. В ролі ядра клієнта, для тестування системи, було обрано мікроконтролер ESP32, який оснащений Wi-Fi модулем, що значно полегшує процес тестування.

Відповідно до написаного для Raspberry Pi коду, було сформовано файл для мікроконтролера, який показано в лістингу 1 додатку В. Цей код дозволить перевірити основні аспекти роботи системи.

Розробка алгоритму роботи системи потребувала деяких досліджень, що стосуються роботи з мікрофоном та локальним сервером інструментами мови програмування Python і в результаті було написано скрипт, що дозволить реалізувати функції, передбачені поставленим завданням

3.3 Тестування системи

Основним етапом тестування є не перевірка роботи самої системи, а підготовка вимог, очікуваних та реальних результатів.

Важливим є формування вимог щодо умов тестування:

– Всі фізичні компоненти підключені до живлення та активні.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						41
Зм.	Арк.	№ докум.	Підпис	Дата		

- Відсутні шуми, яких зазвичай немає в звичайній квартирі або будинку.
- Користувач знаходиться на допустимій відстані від системи.
- Команди проголошуються без особливої чіткості.

Всі ці умови дозволяють забезпечити умови, близькі до тих, які будуть в кінцевого користувача.

Цілями тестування є:

- Перевірка процесу запуску системи на можливі збої.
- Перевірка точності розпізнавання голосових команд.
- Реакція системи на людський фактор: недостатньо чітко вимовлена команда, занадто швидкий або повільний темп вимови, зміни тону голосу.

Процес тестування передбачає перевірку ключових аспектів у роботі всієї системи, але для зручності розуміння результатів, отриманих протягом тестування їх прийнято розділяти на окремі, незалежні частини: «Об'єкт тестування». Він являє собою певну частину системи, що тестується. В рамках проєкту об'єктами тестування будуть:

- Скрипт, який передбачає перевірку коректності запуску написаного програмного коду.
- Мікрофон та модель розпізнавання мовлення, де буде перевірятись точність розпізнавання голосу.
- Функція обробки команд є об'єктом в якому передбачається перевірка коректності формування вихідних команд.
- Сервер та клієнт буде заключним етапом тестування, де перевірятиметься відправлення інформації на підключений до сервера пристрій та її виведення.

Відповідно до поставлених вимог, далі буде описано кожен з передбачених кроків тестування.

Готовність системи для роботи, на етапі тестування, можна перевірити за допомогою виводу в командний рядок статусу моделі нейронної мережі та сервера (рис. 3.4).

					КР.КІ 24.531.04.000 ПЗ	Арк.
						42
Зм.	Арк.	№ докум.	Підпис	Дата		

```

LOG (VoskAPI:ReadDataFiles():model.cc:213) Decoding params beam=13 max-active=7000 lattice-beam=6
LOG (VoskAPI:ReadDataFiles():model.cc:216) Silence phones 1:2:3:4:5
LOG (VoskAPI:RemoveOrphanNodes():nnet-nnet.cc:948) Removed 0 orphan nodes.
LOG (VoskAPI:RemoveOrphanComponents():nnet-nnet.cc:847) Removing 0 orphan components.
LOG (VoskAPI:ReadDataFiles():model.cc:248) Loading i-vector extractor from model/ivector/final.ie
LOG (VoskAPI:ComputeDerivedVars():ivector-extractor.cc:183) Computing derived variables for iVector extractor
LOG (VoskAPI:ComputeDerivedVars():ivector-extractor.cc:204) Done.
LOG (VoskAPI:ReadDataFiles():model.cc:279) Loading HCLG from model/graph/HCLG.fst
LOG (VoskAPI:ReadDataFiles():model.cc:297) Loading words from model/graph/words.txt
LOG (VoskAPI:ReadDataFiles():model.cc:308) Loading winfo model/graph/phones/word_boundary.int
Werkzeug appears to be used in a production deployment. Consider switching to a production web server instead.
* Serving Flask app 'test'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server
* Running on http://127.0.0.1:5000

```

Рисунок 3.4 – Вдалиий запуск системи та сервера

Система готова до роботи, а це означає, що перший об'єкт тестування працює коректно.

Наступним кроком є перевірка точності моделі, послідовне виведення результатів розпізнавання показано на рисунку 3.5.

```

['увімкнув', 'світло']
['вимкнути', 'телевізор']
['хороше', 'сьогодні', 'погода']

```

Рисунок 3.5 – Результат розпізнавання мовлення

Попри певні неточності, результати розпізнавання є прийнятними для їх подальшої обробки. Отже, тестування мікрофона та моделі розпізнавання мовлення пройшло успішно.

Далі було перевірено результат функції перетворення тексту в команди. Очікуваним є виведення умовного коду, наприклад «10» – увімкнення освітлення, отримані результати показано на рисунку 3.6.

```

['увімкнув', 'світло']
10
['вимкнути', 'телевізор']
01
['хороше', 'сьогодні', 'погода']

```

Рисунок 3.6 – Результат обробки тексту

Тести, що стосуються обробки команд було виконано, отримані результати можна вважати прийнятним для їх подальшого відправлення.

Останнім з передбачених тестів є перевірка отримання клієнтом команд. Тобто, команда сформована на сервері повинна передаватись у вигляді HTTP-запиту в локальній мережі. Результат запиту на отримання команди з сервера клієнтом показано на рисунку 3.7.

```
Connected to the server
Command from server: Turning on light
Command from server: Turning off TV
```

Рисунок 3.7 – Відповідь клієнта на отримані команди.

Для зручного представлення інформації про тестування, було складено таблицю в якій можна побачити, що було протестовано, які дії виконано, очікувані результати та відповідність отриманих результатів до очікуваних (таблиця Г.1).

Загалом, це були основні частини проєкту де могли виникнути збої, але в результаті тестування стало зрозумілим, що вони працюють достатньо добре, щоб забезпечити виконання запланованих функцій.

4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ

4.1 Аналіз ринку

Голосові помічники на основі штучного інтерфейсу – це сфера в якій останні роки стрімко розвиваються провідні ІТ-компанії світу. Системи розпізнавання мови, такі як Google Assistant, Amazon Alexa та Apple Siri, використовуються на повсякденному рівні багатьма користувачами з усього світу, що створює попит на подібні рішення і на локальному ринку.

На сьогоднішній день україномовні користувачі мають обмежений доступ до існуючих рішень для голосового управління, що відповідає місцевим контекстам та особливостям. З'являється необхідність у відповідних інструментах, які підтримують українську мову і враховують особливості мовлення та культурні контексти, яку досі в повній мірі не задовільняють великі компанії.

Основним рішенням для україномовного користувача можна вважати розробки компанії Google Nest, розумні динаміки якої працюють на основі голосового помічника Assistant, а від недавня його було доповнено мовною моделлю Gemini, що дозволяє динамічно перекладати будь-які голосові команди, проте переклад не завжди є актуальним або точним, що ще раз доводить актуальність розроблювальної системи.

Отже, основними рішеннями на світовому ринку, які є близькими до розробленого пристрою є:

- Google Nest розумні динаміки від цього виробника підтримують українську мову. Вартість починається від 2800 грн.
- HomePod, що є рішенням від компанії Apple, вартість починається від 5000 грн.
- Echo Dot від компанії Amazon, надає обмежений функціонал для регіону України, але ціна починається від 2000 грн.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						45
Зм.	Арк.	№ докум.	Підпис	Дата		

4.2 Розрахунок витрат на проектування

В процесі розробки системи стало відомо які конкретно компоненти є необхідними для забезпечення запланованого функціоналу, а це дозволяє розписати кошторис всієї системи (таблиця 4.1).

Таблиця 4.1 – Витрати на складові системи

Компонент	Ціна
Raspberry Pi model 4B	4000 грн.
Блок живлення	300-500 грн.
Кабель type-C	100 грн.
Мікрофон з підключенням через Mini jack 3.5mm порт	500-2500 грн.
Загальна вартість:	4900-7100 грн.

Отже, станом на червень 2024 року вартість компонентів проєкту складає, в середньому, 6000 грн, що є допустимим для подібних рішень.

Продуктивність і мотивація до роботи залежать переважно від матеріальних стимулів. Розмір заробітної плати залежить від обсягу виконуваної роботи, умов праці, її складності, результатів виконуваної роботи, особистих якостей і ділової активності компанії.

Заробітна плата поділяється на основну та додаткову. До основної заробітної плати належать винагорода за виконану роботу відповідно до встановлених норм праці (витрати часу, послуги, результати та обов'язки).

Основна заробітна плата затверджується у вигляді окладу за формулою працівника.

До додаткових виплат належать винагорода за надурочну роботу, успіхи, винахідливість та особливі умови праці, у тому числі. Включає в себе надбавки, премії, компенсації та вклади, різні додаткові виплати, заохочення і компенсації.

Заробітна плата повинна виплачуватися працівникам на регулярній основі і не повинна бути меншою за мінімальну заробітну плату, встановлену законом.

Відповідно до Закону України "Про Державний бюджет України на 2022 рік" величина прожиткового мінімуму на одну особу на місяць встановлюється в розмірах 2481 грн. (на 1 січня), мінімальна зарплата - 6700 грн. (на 1 січня), погодинна заробітна плата становить 40,46 грн.

Для розробки системи необхідний 1 інженер та 1 науковий співробітник відповідно до цього було проведено розрахунок заробітної плати, який наведено в таблиці 4.2.

Таблиця 4.2 – Розрахунок заробітної плати проєктувальників

№ п/п	Посада виконавця	Оклад, грн/міс	Відрахування грн/міс	Кількість		Сума з/п, грн.
				чол.	місяців	
1	Інженер	3825	745	1	5	15400
2	Старший науковий співробітник	5423	1057	1	5	21830
		Усього заробітної плати:				37230

Можливий запуск масового виробництва системи розпізнавання голосових команд українською мовою передбачає детальний витрат на проєктування, тому було складено відповідну таблицю, з врахуванням того, що це завдання можна покласти на одну людину (таблиця 4.3).

Відрахування на соціальні потреби працівника становлять 8190 грн. (20% від загальної заробітної плати)

Інші прямі витрати становлять 5500 грн., які були необхідними для покупки мікрокомп'ютера (4000 грн.) та мікрофона (1500 грн.).

Накладні витрати становлять 30% від суми відрахувань на соціальні потреби та прямих витрат – 4107 грн. (13690 грн. * 0.3).

Обсяг планових накопичень було розраховано наступним чином:

$$(13690 \text{ грн.} + 4107 \text{ грн.}) * 0.25 = 4449 \text{ грн.}$$

Кошторисна вартість проєкту:

13690 грн. + 4107 грн. + 4449 грн. = 22246 грн.

Податок на додану вартість становитиме 4449 грн. (22246 грн. * 0.2).

Отже, договірна ціна дорівнює 26695 грн. (22246 грн. + 4449 грн.).

Таблиця 4.2 – Кошторис витрат на проєктування

Найменування статей витрат	Сума витрат, грн	Обґрунтування
Зарплата проєктувальників	37230	
Контрагентські роботи та послуги	-	Не проводились
Витрати на відрядження	-	Не проводились
Відрахування на соціальні потреби	8190	
Витрати на складові системи	5500	
Усього прямих витрат	13690	
Накладні витрати	4107	
Планові накопичення	4449	
Кошторисна вартість проєкту	22246	
Податок на додану вартість	4449	
Договірна ціна розробки	26695	

4.3 Обґрунтування доцільності проєкту

Розробка автономної системи розпізнавання голосових команд, спрямованої на україномовних користувачів, має потенціал вийти на ринок з перевагами у формуванні персоналізованих рішень, які відповідають специфічним потребам користувачів в Україні.

Потенційна аудиторія включає україномовних користувачів з різних галузей, таких як освіта, бізнес, медицина та побутове використання. Важливо враховувати їхні потреби та очікування від голосових інтерфейсів для успішного впровадження продукту на ринок.

З урахуванням розвитку мобільних технологій та "розумних" пристроїв, які активно використовуються в Україні, існує значний потенціал для впровадження автономних систем розпізнавання голосових команд. Це відкриває можливості для створення нових продуктів і сервісів, які підтримують українську мову та відповідають специфічним потребам місцевих користувачів.

З впровадженням автономних систем розпізнавання голосових команд українською мовою можливе створення інноваційних мобільних додатків для управління побутовими пристроями.

Таким чином, розвиток автономних систем розпізнавання голосових команд українською мовою відкриває широкі можливості для інновацій та покращення якості життя користувачів, розширюючи спектр їхніх можливостей і забезпечуючи відповідність місцевим потребам і очікуванням.

За допомогою голосових команд користувачі зможуть контролювати температуру та вологість у приміщенні, що забезпечить оптимальні умови для життя і роботи. Крім того, можливість управління розумними електроприладами через голосові команди значно спростить їхнє використання, зменшуючи час і зусилля, потрібні для керування кожним пристроєм окремо.

Такий підхід не лише підвищить комфорт користувачів, але й дозволить зменшити витрати енергії та ресурсів завдяки більш ефективному управлінню побутовими системами. Впровадження цих технологій також сприятиме екологічній стійкості, зменшуючи вплив на довкілля через раціональне використання енергетичних ресурсів.

Отже, впровадження автономних систем розпізнавання голосових команд українською мовою не лише відкриває нові можливості для користувачів у їхньому повсякденному житті, але й сприяє збереженню ресурсів та підвищенню якості життя в умовах сучасного технологічного прогресу.

ВИСНОВКИ

Тема голосових асистентів та ШІ активно просовується в маси останні 10 років, а провідні IT-компанії розробляють власні рішення в цій сфері. Google, Amazon та Apple є основними постачальниками подібних продуктів на світовому ринку.

Для розробки проєкту необхідним було ознайомлення з різними сферами:

- Робота з мікрокомп'ютерами.
- Обробка звуку.
- Робота та використання сучасних моделей розпізнавання мови.

В процесі розробки було виконано досить комплексне завдання, яке складалось з наступних пунктів:

- Детального аналізу можливих варіантів вирішення проблеми, де було проведено дослідження наявних рішень, що дозволило сформулювати чіткі завдання для проєктування системи.

- Моделювання майбутньої системи, що є невід'ємною частиною будь-якого проєкту.

- Розробка системи та її подальше тестування стало повноцінним підсумком всіх знань, отриманих протягом навчання. Окрім цього, було використано велику кількість інформації про теми, дотичні до проєкту.

- Детальний аналіз техніко-економічних аспектів був необхідним для ознайомлення з основами економічної частини IT-проєкту.

Результатом роботи стало набуття великої кількості досвіду та корисних навичок які стосуються роботи з нейронними мережами та аудіо-інформацією.

Фінальна версія проєкту відповідає всім вимогам які було поставлено на етапах аналізу та моделювання системи.

					КР.КІ 24.531.04.000 ПЗ	Арк.
						50
Зм.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Основна інформація про динаміки Google Nest : веб-сайт. URL: <https://www.androidauthority.com/what-is-google-nest-3232791/> (дата звернення: 10.03.2024).
2. Основна інформація про динаміки HomePod : веб-сайт. URL: <https://www.intego.com/mac-security-blog/how-to-do-everything-with-the-homepod/> (дата звернення: 10.03.2024).
3. Основна інформація про динаміки Echo Dot 5 : веб-сайт. URL: <https://www.cnet.com/home/smart-home/amazon-echo-dot-5th-gen-review-the-best-echo-on-a-budget/> (дата звернення: 10.03.2024).
4. Проектування схем : веб-сайт. URL: <https://www.cnet.com/home/smart-home/amazon-echo-dot-5th-gen-review-the-best-echo-on-a-budget/> (дата звернення: 10.03.2024).
5. Розпізнавання мовлення : веб-сайт. URL: https://uk.wikipedia.org/wiki/Розпізнавання_мовлення (дата звернення: 27.04.2024).
6. Документація для встановлення та використання бібліотеки vosk : веб-сайт. URL: <https://alphacephei.com/vosk/> (дата звернення: 05.05.2024).
7. Завантаження vosk-тоделі : веб-сайт. URL: <https://alphacephei.com/vosk/models> (дата звернення: 15.05.2024).
8. Методичні рекомендації до виконання кваліфікаційної роботи для студентів спеціальності 123 «Комп'ютерна інженерія» освітньої програми «Інженерія Інтернету речей» / Павлюс В.П., Посвятовська О.Б., Кульчинська Н.З. – Галицький фаховий коледж імені В'ячеслава Чорновола, Тернопіль, 2023. 52с.

ДОДАТКИ

Додаток А. Блок-схема алгоритму роботи системи

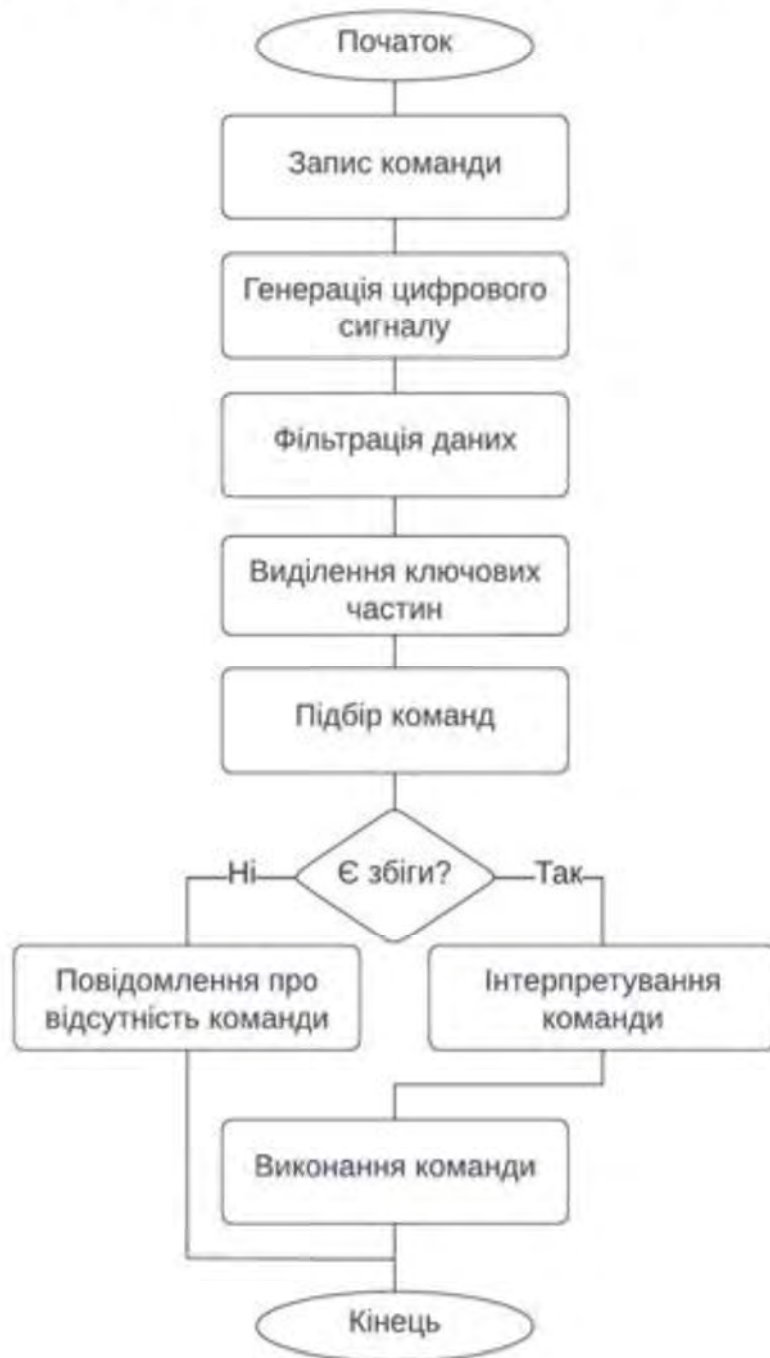


Рисунок 1

Додаток Б. Функція формування команди

Лістинг програмного коду

```
def process_voice_command(command):  
    text = command.split()  
    to_send = ""  
  
    if len(text) >= 1:  
        if text[0].startswith("вкл") or text[0].startswith("увім")  
or text[0].startswith("ввим"):   
            to_send += "1"  
            if text[1].startswith("сві") or  
text[1].startswith("осв"):   
                to_send += "0"  
            elif text[1].startswith("тел"):   
                to_send += "1"  
            elif text[1].startswith("кон"):   
                to_send += "2"  
  
        elif text[0].startswith("викл") or  
text[0].startswith("вимк"):   
            to_send += "0"  
            if text[1].startswith("сві") or  
text[1].startswith("осв"):   
                to_send += "0"  
            elif text[1].startswith("тел"):   
                to_send += "1"  
            elif text[1].startswith("кон"):   
                to_send += "2"  
  
    return to_send
```


Лістинг програмного коду

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

const char* ssid = "SSID";
const char* password = "PASSWORD";

const char* serverName = "http://127.0.0.1:5000/data";

WiFiClient client;
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup() {
    Serial.begin(115200);
    lcd.init();
    lcd.backlight();
    WiFi.begin(ssid, password);

    // Перевірка з'єднання з Wi-Fi мережею
    while (WiFi.status() != WL_CONNECTED) {
        delay(1000);
        Serial.println("Connecting to WiFi...");
    }
    Serial.println("Connected to WiFi");
}

void loop() {
    if (WiFi.status() == WL_CONNECTED) {
        HTTPClient http;
        String serverPath = String(serverName);
```

```

http.begin(client, serverPath);
int httpResponseCode = http.GET();

if (httpResponseCode > 0) {
    String response = http.getString();
    Serial.println(httpResponseCode);
    Serial.println(response);

    // Перетворення відповіді на команду
    handleResponse(response);

} else {
    Serial.print("Error on sending GET: ");
    Serial.println(httpResponseCode);
}

http.end();
}
delay(10000);
}

void handleResponse(String response) {
    if (response.length() >= 2) {
        char firstChar = response.charAt(0);
        char secondChar = response.charAt(1);

        if (firstChar == '1') {
            if (secondChar == '0') {
                lcd.clear();
                lcd.print("Turning on light");
            } else if (secondChar == '1') {
                lcd.clear();
                lcd.print("Turning on TV");
            } else if (secondChar == '2') {

```



```

        lcd.clear();
        lcd.print("Turning on AC");
    }
} else if (firstChar == '0') {
    if (secondChar == '0') {
        lcd.clear();
        lcd.print("Turning off light");
    } else if (secondChar == '1') {
        lcd.clear();
        lcd.print("Turning off TV");
    } else if (secondChar == '2') {
        lcd.clear();
        lcd.print("Turning off AC");
    }
} else {
    lcd.clear();
    lcd.print("Unknown command");
}
}
}

```

Додаток Г. Таблиця результатів тестування

Таблиця 1

Об'єкт тестування	№	Дія	Очікуваний результат	Отриманий результат
Програма	1	Запуск програми	Інформація про запуск моделі	Інформація про запуск моделі
	2	Запуск сервера	Адреса сервера	Адреса сервера
Мікрофон та модель розпізнавання мови	3	Сказати «увімкни світло»	увімкни світло	увімкнув світло
	4	Сказати «вимкни телевізор»	вимкни телевізор	вимкнути телевізор
	5	Сказати «хороша сьогодні погода»	хороша сьогодні погода	хороше сьогодні погода
Функція обробки команд	6	Повторити дію №3	10	10
	7	Повторити дію №4	02	02
	8	Повторити дію №5	-	-
Сервер та клієнт	9	Повторити дію №3	Turning on light	Turning on light
	10	Повторити дію №5	Unknown command	Unknown command