



THE
UNIVERSITY OF
LAHORE

Mobile & Tech- AI Gadget Shop

(AI Chatbot for Shop)

Submitted By:

Muhammad Ahmed	(70147999)
Husnain Mavia	(70147247)
Muhammad Bilal	(70150794)
Muzammil Ali	(70145807)
Ahmad Taj	(70145806)

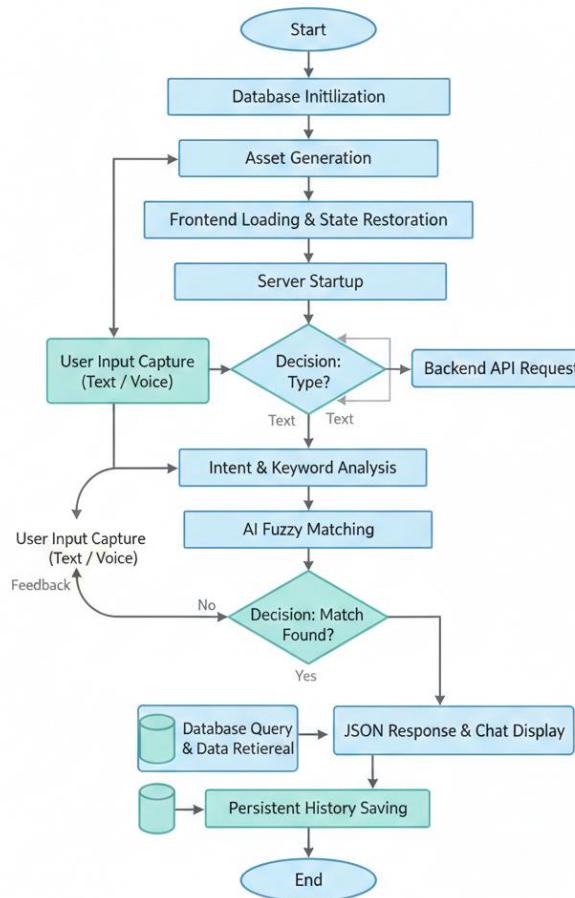
Submitted To:

- Dr. Syed Hamedoon

1) Project Description

AI Gadget Shop is an AI-powered e-commerce web application developed to provide instant product information, which reduces the workload on shop staff by addressing customer inquiries automatically. The system utilizes a Flask backend and a MySQL database to handle requests and store structured data regarding smartphones, accessories, and gadgets. By employing Fuzzy Matching and Natural Language Processing, the chatbot can understand user queries and accurately retrieve details such as prices, availability, and categories from a predefined dataset. The frontend is built using HTML, CSS, and JavaScript to offer a modern interface that features voice-based interaction and maintains a persistent chat history for a seamless user experience. This project modernizes shop operations by ensuring that customers receive quick and accurate information without the delays often caused by limited staff availability.

2) Project Flow Diagram



3) Project Code

HTML:

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>AI Gadget Shop - V3 (FINAL)</title>

    <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600;700&display=swap" rel="stylesheet">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">

    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}>
</head>

<body>

    <nav>
        <div class="logo">Mobile and Tech</div>
        <div class="nav-links">
            <a href="#">Home</a>
            <a href="#smartphones">Smartphones</a>
            <a href="#accessories">Accessories</a>
            <a href="#gadgets">Gadgets</a>
        </div>
    </nav>

    <section class="hero">
        <div class="hero-content">
            <h1>Experience the Future of Tech</h1>
            <p class="subtitle">Discover minimal aesthetics, premium smartphones, and next-gen gadgets.</p>
            <button class="btn-main">Shop Now</button>
        </div>
    </section>

    <div id="smartphones" class="features">
```

```
<h2 style="width: 100%; text-align: center; margin-bottom: 20px;">Latest Samsung Smartphones</h2>
<div class="feature-card">
    
    <h3>Samsung Galaxy S23</h3>
    <p>Flagship performance & camera.</p>
</div>
<div class="feature-card">
    
    <h3>Samsung Galaxy A34</h3>
    <p>Balanced performance, vivid display.</p>
</div>
<div class="feature-card">
    
    <h3>Samsung Galaxy A73</h3>
    <p>108MP camera, sleek design.</p>
</div>
</div>

<!-- Products: Accessories -->
<div id="accessories" class="features">
    <h2 style="width: 100%; text-align: center; margin-bottom: 20px;">Premium Accessories</h2>
    <div class="feature-card">
        
        <h3>Fast Chargers</h3>
        <p>Power up your device in minutes.</p>
    </div>
    <div class="feature-card">
        
        <h3>Wireless Earbuds</h3>
        <p>Noise cancellation & deep bass.</p>
    </div>
    <div class="feature-card">
        
        <h3>Smart Watches</h3>
        <p>Track your health & notifications.</p>
    </div>
</div>

<!-- Products: Gadgets -->
<div id="gadgets" class="features">
    <h2 style="width: 100%; text-align: center; margin-bottom: 20px;">Smart Gadgets</h2>
    <div class="feature-card">
```

```

<h3>Drone Cameras</h3>
<p>Capture the world from above.</p>
</div>
<div class="feature-card">
    
    <h3>VR Headsets</h3>
    <p>Immersive virtual reality gaming.</p>
</div>
<div class="feature-card">
    
    <h3>Gaming Consoles</h3>
    <p>Next-gen gaming experience.</p>
</div>
</div>

<div class="chat-toggle-btn" onclick="toggleChat()">
    <span class="chat-icon">●</span>
</div>

<div class="chat-container" id="chatContainer">
    <div class="chat-header">
        <span>AI Assistant</span>
        <div style="display: flex; gap: 10px;">
            <button class="close-btn" title="Clear Chat"
onlick="clearChat()">
                <i class="fa-solid fa-trash"></i>
            </button>
            <button class="close-btn" onclick="toggleChat()">
                <i class="fa-solid fa-xmark"></i>
            </button>
        </div>
    </div>
    <div id="chat-box">
        <div class="message bot">
            Hey there! 🙋 Welcome to Mobile and Tech. <br> I can help you
check prices, stock, colors, and warranty for any product!
        </div>
    </div>
    <div class="input-area">
        <input type="text" id="userInput" placeholder="Type a message...">
        <button id="micBtn" onclick="toggleVoice()">
            <i class="fa-solid fa-microphone"></i>
        </button>
    </div>
</div>
```

```
<button onclick="sendMessage()">
    <i class="fa-solid fa-paper-plane"></i>
</button>
</div>
</div>

<script src="{{ url_for('static', filename='js/script.js') }}></script>
</body>

</html>
```

```
CSS: :root {
    --primary-color: #00f2fe;
    --secondary-color: #4facfe;
    --bg-dark: #0f172a;
    --glass-bg: rgba(255, 255, 255, 0.1);
    --glass-border: rgba(255, 255, 255, 0.2);
    --text-color: #ffffff;
}

body {
    font-family: 'Poppins', sans-serif;
    margin: 0;
    padding: 0;
    background-color: var(--bg-dark);
    color: var(--text-color);
    overflow-x: hidden;
}

nav {
    display: flex;
    justify-content: space-between;
    align-items: center;
    padding: 20px 50px;
    background: rgba(15, 23, 42, 0.9);
    backdrop-filter: blur(10px);
    position: fixed;
    width: 100%;
    top: 0;
    z-index: 1000;
    box-sizing: border-box;
    border-bottom: 1px solid var(--glass-border);
```

```
}

.logo {
    font-size: 24px;
    font-weight: 700;
    background: linear-gradient(to right, var(--primary-color), var(--secondary-color));
    -webkit-background-clip: text;
    -webkit-text-fill-color: transparent;
}

.nav-links a {
    color: var(--text-color);
    text-decoration: none;
    margin-left: 30px;
    font-weight: 400;
    transition: color 0.3s;
}

.nav-links a:hover {
    color: var(--primary-color);
}

.hero {
    height: 100vh;
    display: flex;
    align-items: center;
    justify-content: center;
    text-align: center;
    background: radial-gradient(circle at center, #1e293b 0%, #0f172a 100%);
    position: relative;
}

.hero::before {
    content: '';
    position: absolute;
    width: 100%;
    height: 100%;
    background: url('https://images.unsplash.com/photo-1550745165-9bc0b252726f?auto=format&fit=crop&q=80') center/cover;
    opacity: 0.1;
    z-index: 0;
}

.hero-content {
```

```
z-index: 1;
max-width: 800px;
padding: 20px;
}

h1 {
  font-size: 60px;
  margin-bottom: 20px;
  line-height: 1.2;
  background: linear-gradient(to right, #fff, #94a3b8);
  -webkit-background-clip: text;
  -webkit-text-fill-color: transparent;
}

p.subtitle {
  font-size: 20px;
  color: #cbd5e1;
  margin-bottom: 40px;
}

.btn-main {
  padding: 15px 40px;
  background: linear-gradient(45deg, var(--secondary-color), var(--primary-color));
  color: white;
  border: none;
  border-radius: 50px;
  font-size: 18px;
  font-weight: 600;
  cursor: pointer;
  box-shadow: 0 10px 20px rgba(0, 242, 254, 0.3);
  transition: transform 0.3s, box-shadow 0.3s;
}

.btn-main:hover {
  transform: translateY(-3px);
  box-shadow: 0 15px 30px rgba(0, 242, 254, 0.5);
}

.features {
  padding: 60px 20px;
  display: flex;
  justify-content: center;
  flex-wrap: wrap;
  gap: 30px;
}
```

```
    max-width: 1200px;
    margin: 0 auto;
}

.feature-card {
    background: var(--glass-bg);
    border: 1px solid var(--glass-border);
    padding: 20px;
    border-radius: 20px;
    text-align: center;
    transition: transform 0.3s;
    width: 300px;
    display: flex;
    flex-direction: column;
    align-items: center;
}

.feature-card img {
    width: 100%;
    height: 200px;
    object-fit: cover;
    border-radius: 15px;
    margin-bottom: 15px;
    box-shadow: 0 5px 15px rgba(0, 0, 0, 0.3);
}

.feature-card:hover {
    transform: translateY(-10px);
    background: rgba(255, 255, 255, 0.15);
}

.feature-card h3 {
    margin-top: 20px;
    color: var(--primary-color);
}

.chat-toggle-btn {
    position: fixed;
    bottom: 30px;
    right: 30px;
    width: 60px;
    height: 60px;
    background: linear-gradient(135deg, var(--primary-color), var(--secondary-color));
    border-radius: 50%;
}
```

```
display: flex;
justify-content: center;
align-items: center;
cursor: pointer;
box-shadow: 0 10px 25px rgba(0, 0, 0, 0.5);
z-index: 2000;
transition: transform 0.3s;
animation: bounce 2s infinite;
}

.chat-toggle-btn:hover {
    transform: scale(1.1);
}

.chat-icon {
    font-size: 30px;
    color: white;
}

.chat-container {
    position: fixed;
    bottom: 100px;
    right: 30px;
    width: 380px;
    height: 500px;
    background: rgba(25, 35, 60, 0.95);
    backdrop-filter: blur(20px);
    border: 1px solid var(--glass-border);
    border-radius: 20px;
    box-shadow: 0 20px 50px rgba(0, 0, 0, 0.5);
    display: flex;
    flex-direction: column;
    overflow: hidden;
    z-index: 2000;
    opacity: 0;
    transform: translateY(20px) scale(0.9);
    visibility: hidden;
    transition: all 0.3s cubic-bezier(0.165, 0.84, 0.44, 1);
}

.chat-container.active {
    opacity: 1;
    transform: translateY(0) scale(1);
    visibility: visible;
}
```

```
.chat-header {  
    background: linear-gradient(to right, var(--secondary-color), var(--primary-color));  
    padding: 15px;  
    color: white;  
    font-weight: bold;  
    display: flex;  
    justify-content: space-between;  
    align-items: center;  
}  
  
.close-btn {  
    cursor: pointer;  
    font-size: 20px;  
    background: none;  
    border: none;  
    color: white;  
}  
  
#chat-box {  
    flex: 1;  
    padding: 20px;  
    overflow-y: auto;  
    display: flex;  
    flex-direction: column;  
    gap: 15px;  
}  
  
.message {  
    padding: 10px 15px;  
    border-radius: 15px;  
    max-width: 80%;  
    font-size: 14px;  
    line-height: 1.5;  
    animation: fadeIn 0.3s ease;  
}  
  
.bot {  
    background: rgba(255, 255, 255, 0.1);  
    color: #eee;  
    align-self: flex-start;  
    border-bottom-left-radius: 2px;  
}
```

```
.user {
    background: var(--secondary-color);
    color: white;
    align-self: flex-end;
    border-bottom-right-radius: 2px;
}

.input-area {
    padding: 15px;
    background: rgba(0, 0, 0, 0.3);
    display: flex;
    gap: 10px;
}

.input-area input {
    flex: 1;
    padding: 12px;
    border-radius: 25px;
    border: 1px solid var(--glass-border);
    background: rgba(255, 255, 255, 0.05);
    color: white;
    outline: none;
}

.input-area input::placeholder {
    color: rgba(255, 255, 255, 0.5);
}

.input-area button {
    background: var(--primary-color);
    border: none;
    width: 40px;
    height: 40px;
    border-radius: 50%;
    color: #0f172a;
    font-weight: bold;
    cursor: pointer;
    display: flex;
    align-items: center;
    justify-content: center;
}

@keyframes bounce {
    0%, 100% { transform: translateY(0); }
    50% { transform: translateY(-10px); }
}
```

```

}

@keyframes fadeIn {
    from { opacity: 0; transform: translateY(10px); }
    to { opacity: 1; transform: translateY(0); }
}

@keyframes pulse {
    0% { transform: scale(1); box-shadow: 0 0 0 0 rgba(0, 168, 132, 0.7); }
    70% { transform: scale(1.1); box-shadow: 0 0 0 10px rgba(0, 168, 132, 0); }
    100% { transform: scale(1); box-shadow: 0 0 0 0 rgba(0, 168, 132, 0); }
}

```

JavaScript:

```

function toggleChat() {
    const container = document.getElementById('chatContainer');
    container.classList.toggle('active');
    if (container.classList.contains('active')) {
        setTimeout(() => document.getElementById('userInput').focus(), 300);
    }
}

function clearChat() {
    const chatBox = document.getElementById('chat-box');
    chatBox.innerHTML = ` <div class="message bot">Hey there ! 🙌 Welcome to
Mobile and Tech. <br>I can help you check prices, stock, colors, and warranty for
any product ! </div>`;
}

function handleKeyPress(e) {
    if (e.key === 'Enter') {
        sendMessage();
    }
}

async function sendMessage() {
    const input = document.getElementById('userInput');
    const chatBox = document.getElementById('chat-box');
    const message = input.value.trim();

    if (!message) return;

```

```

chatBox.innerHTML += `<div class="message user">${message}</div>`;
input.value = '';
chatBox.scrollTop = chatBox.scrollHeight;

try {
    const response = await fetch('/chatbot', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify({ message: message })
    });
    const data = await response.json();

    let botMessage = `<div class="message bot">${data.reply}</div>`;
    if (data.image_url) {
        botMessage += `<br><br>`;
    }
    botMessage += `</div>`;
    chatBox.innerHTML += botMessage;

    speakText(data.reply.replace(<[^>]*>/g, ''));

} catch (error) {
    chatBox.innerHTML += `<div class="message bot" style="color:#ff6b6b">⚠️ Error: Backend not connected !</div>`;
}

chatBox.scrollTop = chatBox.scrollHeight;
}

const micBtn = document.getElementById('micBtn');
const SpeechRecognition = window.SpeechRecognition ||
window.webkitSpeechRecognition;
const recognition = SpeechRecognition ? new SpeechRecognition() : null;

if (recognition) {
    recognition.continuous = false;
    recognition.lang = 'en-US';
    recognition.interimResults = false;

    recognition.onstart = function () {
        micBtn.innerHTML = '<i class="fa-solid fa-microphone-lines"></i>';
        micBtn.style.animation = "pulse 1.5s infinite";
    }
}

```

```
document.getElementById('userInput').placeholder = "Listening... Speak now!";
};

recognition.onsoundstart = function () {
    document.getElementById('userInput').placeholder = "Sound detected...";
};

recognition.onspeechend = function () {
    micBtn.innerHTML = '<i class="fa-solid fa-spinner fa-spin"></i>';
    micBtn.style.animation = "none";
    document.getElementById('userInput').placeholder = "Processing speech...";
};

recognition.onend = function () {
    micBtn.innerHTML = '<i class="fa-solid fa-microphone"></i>';
    micBtn.style.animation = "none";
    if
(document.getElementById('userInput').placeholder.includes("Processing")) {
        document.getElementById('userInput').placeholder = "Did not catch that. Try again.";
    } else {
        document.getElementById('userInput').placeholder = "Type a message...";
    }
};

recognition.onresult = function (event) {
    const transcript = event.results[0][0].transcript;
    document.getElementById('userInput').value = transcript;
    sendMessage();
};

recognition.onerror = function (event) {
    micBtn.innerHTML = '<i class="fa-solid fa-microphone-slash"></i>';
    micBtn.style.animation = "none";
    console.error("Speech Recognition Error:", event.error);
};

} else {
    if (micBtn) micBtn.style.display = 'none';
    console.log("Web Speech API not supported in this browser.");
}
```

```

function toggleVoice() {
  if (recognition) {
    try {
      recognition.start();
    } catch (e) {
      console.error(e);
    }
  } else {
    alert("Your browser does not support Voice Input. Please use Chrome/Edge.");
  }
}

function speakText(text) {
  const synth = window.speechSynthesis;
  if (!synth) return;

  if (synth.speaking) {
    synth.cancel();
  }

  const utterance = new SpeechSynthesisUtterance(text);
  utterance.lang = 'en-US';
  utterance.rate = 1;
  utterance.pitch = 1;

  synth.speak(utterance);
}

```

Backend:

```

import pymysql
from flask import Flask, request, jsonify, render_template
from flask_cors import CORS
from fuzzywuzzy import process

app = Flask(__name__)
app.config['TEMPLATES_AUTO_RELOAD'] = True
CORS(app)

db_config = {
  "host": "127.0.0.1",

```

```
"user": "root",
"password": "12345678",
"database": "MyAIGadgetShop"
}

def get_db_connection():

    try:
        return pymysql.connect(
            host=db_config["host"],
            user=db_config["user"],
            password=db_config["password"],
            database=db_config["database"],
            cursorclass=pymysql.cursors.DictCursor
        )
    except Exception as e:
        print(f"Database Connection Error: {e}")
        return None

def get_product_details(user_msg):

    conn = get_db_connection()
    if not conn:
        return "Database connection error.", None

    cursor = conn.cursor()

    products_map = {}
    tables = ['Smartphones', 'Accessories', 'SmartGadgets']

    for table in tables:
        cursor.execute(f"SELECT name FROM {table}")
        for row in cursor.fetchall():
            products_map[row['name']] = table

    all_product_names = list(products_map.keys())

    best_match, score = process.extractOne(user_msg, all_product_names)

    response_text = ""
    image_url = None

    if score > 55:
        table_name = products_map[best_match]
```

```

        cursor.execute(f"SELECT * FROM {table_name} WHERE name = %s",
(best_match,))
        product = cursor.fetchone()

    if product:
        name = product['name']
        price = int(product['price'])
        stock = product['stock']
        brand = product['brand']
        color = product.get('color', 'N/A')
        image_url = product['image_url']

        extra_info = ""
        if 'warranty' in product:
            extra_info = f"<br>Warranty: {product['warranty']}"
        elif 'battery_life' in product:
            extra_info = f"<br>Battery Life: {product['battery_life']}"

        response_text = (f"<b>Found: {name}</b> ({brand})<br>"
                        f"Price: PKR {price}<br>"
                        f"Stock: {stock}<br>"
                        f"Color: {color}"
                        f"{extra_info}")
    else:
        response_text = "Product details not found in database."
    else:
        response_text = "I couldn't find a matching product. Please try asking
for 'S23', 'Headphones', or other gadgets."

    conn.close()
    return response_text, image_url

# --- Routes ---

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/chatbot', methods=['POST'])
def chatbot():
    """API endpoint for the AI Chatbot."""
    data = request.json
    user_msg = data.get("message", "")

```

```

    reply, image_url = get_product_details(user_msg)

    return jsonify({
        "reply": reply,
        "image_url": image_url
    })

if __name__ == '__main__':
    print("AIGadgetShop Backend Started. AI Fuzzy Matching logic ready.")
    app.run(debug=True, port=5000)

```

Database:

```

import pymysql

def setup_database():
    try:
        # Pehle connection banate hain bina database select kiye
        conn = pymysql.connect(
            host="127.0.0.1",
            user="root",
            password="12345678"
        )
        cursor = conn.cursor()

        # Database create karte hain
        cursor.execute("CREATE DATABASE IF NOT EXISTS MyAIGadgetShop")
        cursor.execute("USE MyAIGadgetShop")

        # Smartphones table
        cursor.execute("DROP TABLE IF EXISTS Smartphones")
        cursor.execute("""
CREATE TABLE Smartphones (
    id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100),
    brand VARCHAR(50),
    price INT,
    stock VARCHAR(20),
    warranty VARCHAR(50),
    color varchar(50)
)
""")
    """

```

```

smartphones_data = [
    ('Samsung Galaxy S23 Ultra', 'Samsung', 320000, 'Available', 'black',
'1 Year Official'),
    ('Samsung Galaxy S22 Plus', 'Samsung', 185000, 'Available', 'white',
'1 Year'),
    ('Samsung Galaxy Z Fold 5', 'Samsung', 550000, 'Available',
'black','1 Year Official'),
    ('Samsung Galaxy Z Flip 5', 'Samsung', 280000, 'Available', 'gray',
'1 Year Official'),
    ('Samsung Galaxy A73', 'Samsung', 145000, 'Available', 'white', '1
Year'),
    ('Samsung Galaxy A54', 'Samsung', 125000, 'Available', 'brown', '1
Year Official'),
    ('Samsung Galaxy A34', 'Samsung', 95000, 'Available', 'black', '1
Year'),
    ('Samsung Galaxy A24', 'Samsung', 72000, 'Available', 'blue', '1
Year'),
    ('Samsung Galaxy M34', 'Samsung', 65000, 'Available', 'black', 'No
Warranty'),
    ('Samsung Galaxy F54', 'Samsung', 88000, 'Out of Stock', 'white', '1
Year')
]
cursor.executemany("INSERT INTO Smartphones (name, brand, price, stock,
color, warranty) VALUES (%s, %s, %s, %s, %s, %s)", smartphones_data)

# Accessories table
cursor.execute("DROP TABLE IF EXISTS Accessories")
cursor.execute("""
CREATE TABLE Accessories (
    id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100),
    brand VARCHAR(50),
    price INT,
    stock VARCHAR(20),
    warranty VARCHAR(50),
    color varchar(50)
)
""")
accessories_data = [
    ('Fast Charger 25W', 'Samsung', 4500, 'Available', 'white', '6
Months'),
    ('USB Type-C Cable', 'Anker', 1200, 'Available', 'black','1 Year'),
    ('Power Bank 20000mAh', 'Belkin', 5500, 'Available', 'black', '1
Year'),
]

```

```

        ('Wired Earphones', 'Apple', 6500, 'Available', 'white', '1 Year
Official'),
        ('Wireless Earbuds', 'Ronin', 2200, 'Available', 'black', '6
Months'),
        ('Mobile Cover', 'Generic', 1500, 'Available',
'black,white,gray,blue', 'No Warranty'),
        ('Wall Charger 18W', 'Baseus', 450, 'Available', 'black', 'No
Warranty'),
        ('Car Charger Dual Port', 'Ugreen', 1800, 'Available', 'gray', '3
Months'),
        ('Wireless Headphones', 'Joyroom', 5200, 'Out of Stock', 'black', '6
Months'),
        ('Screen Protector Tempered Glass', 'Razer', 3800, 'Available',
'black', '1 Year')
    ]
cursor.executemany("INSERT INTO Accessories (name, brand, price, stock,
color, warranty) VALUES (%s, %s, %s, %s, %s, %s)", accessories_data)

# SmartGadgets table
cursor.execute("DROP TABLE IF EXISTS SmartGadgets")
cursor.execute("""
CREATE TABLE SmartGadgets (
    id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100),
    brand VARCHAR(50),
    price INT,
    stock VARCHAR(20),
    battery_life VARCHAR(50),
    color varchar(50)
)
""")
gadgets_data = [
    ('Smart Watch', 'Apple', 125000, 'Available', 'black','18 Hours'),
    ('Bluetooth Speaker', 'Samsung', 75000, 'Available', 'black', '40
Hours'),
    ('Fitness Band', 'Xiaomi', 9500, 'Available', 'gray', '16 Days'),
    ('Fitness Band Pro', 'Audionic', 12000, 'Available', 'blue', '6
Hours'),
    ('AirPods (Original)', 'Generic', 8500, 'Out of Stock','black', '4
Days'),
    ('Smart Glasses Basic', 'Tanix', 6500, 'Available','brown', 'Constant
Power'),
    ('Mini Smart Camera', 'Xiaomi', 8800, 'Available','black', 'Constant
Power'),
]

```

```

        ('AirPods (Copy)', 'Philips', 4200, 'Available', 'white', 'No
Battery'),
        ('Smart Table Lamp', 'Generic', 1200, 'Available', 'black', '3
Hours'),
        ('Digital Smart Scale', 'Huawei', 5500, 'Available', 'blue', '1 Year
Battery')
    ]
cursor.executemany("INSERT INTO SmartGadgets (name, brand, price, stock,
color, battery_life) VALUES (%s, %s, %s, %s, %s, %s)", gadgets_data)

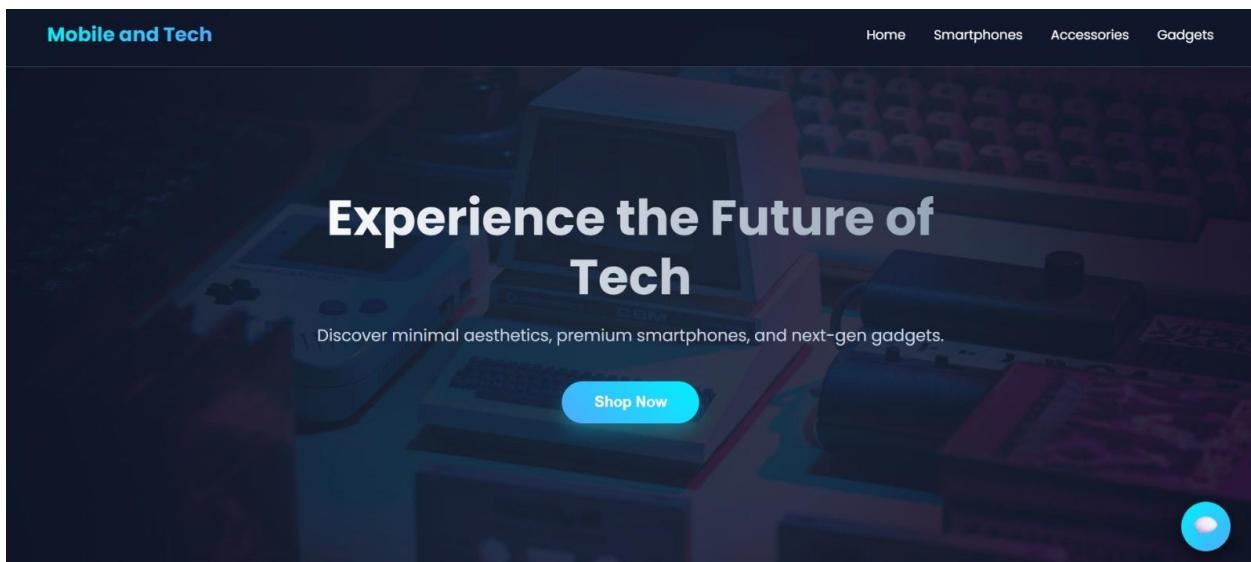
conn.commit()
print("Database MyAIGadgetShop aur tables kamyabi se ban gaye hain!")
conn.close()

except pymysql.MySQLError as err:
    print(f"Error: {err}")
    print("Kya apka XAMPP/MySQL server start hai?")

if __name__ == "__main__":
    setup_database()

```

4) Output (Chatbot)



Premium Accessories**Wireless Earbuds**

Noise cancellation & deep bass.

**Smart Watches**

Track your health & notifications.

**Fast Chargers**

Power up your device in minutes.

**Latest Samsung Smartphones****Samsung S23 Ultra****Samsung Galaxy S23**

Flagship performance & camera.

Galaxy A34**Samsung Galaxy A34**

Balanced performance, vivid display.

AI Assistant

Hey there! 🌟 Welcome to Mobile and Tech.
I can help you check prices, stock, colors, and warranty for any product!

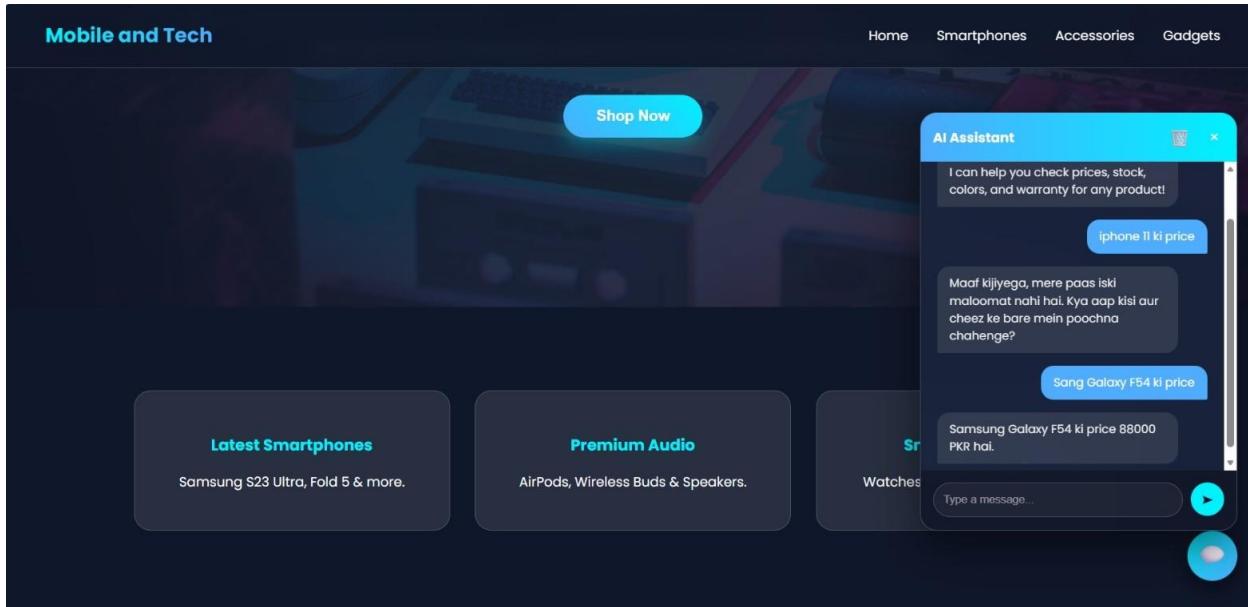
Samsung Galaxy S22 Plus ki price btao

The price of **Samsung Galaxy S22 Plus** is **PKR 185000**.

Samsung S22 Plus

Type a message...





5) Scope

The scope of the **AI Gadget Shop** project is focused on providing a specialized AI assistant that handles customer inquiries based on a specific product dataset. The system is currently capable of delivering instant details regarding product names, prices, brands, stock availability, and category-specific information like warranty or battery life. It also provides essential shop information, such as the physical address, operating hours, and basic FAQs. Technically, the scope includes a modern user interface that supports both text and voice-based interactions while ensuring that the conversation history is saved locally for a continuous user experience. However, the project does not include features for online payments, delivery logistics, or automated stock management. Additionally, the chatbot is strictly limited to the provided dataset and will not respond to queries regarding information outside of its predefined product list.

6) GitHub Project Repository Link

<https://github.com/Muzammil9090/Artificial-Intelligence-Project-Repository>