



PROYECTO (ABP)

Módulo: Programador

Evidencia - 03

Integrantes

- Sergio Rodríguez (abandono)
- Ariel Nicolás Romano
- Luis Nicolas Asensio Lubrano
- Daniel Esteban Gonzalez Lara
- Alan Ezequiel Altamirano (abandono)
- Lucas Ezequiel Cisterna Castro (abandono)

Índice

Actividad Integradora N° 3	3
Programación I	3
Base de datos	3
Ética y Deontología Profesional	4
Entregables	5
Programación I	5
Base de datos.	5
	8
Relaciones Entre Entidades - Modelo relacional	9
Ética y Deontología Profesional	12
Impacto Social y Económico	12
Beneficios para los usuarios	14
Impacto Ambiental	15
Relación con el Pilar de Sostenibilidad del AWS Well-Architected Framework	
Manual de usuario	17

Actividad Integradora N° 3

Programación I

Partiendo del programa de consola desarrollado en la evidencia anterior, modificar el mismo a fin de :

1. Permitir el registro de un nuevo **usuario estándar** e inicio de sesión.
2. Si el usuario estándar inició sesión satisfactoriamente, permitir las siguientes acciones mediante un menú:
 - a. Permitir consultar los datos personales.
 - b. Permitir activar/ejecutar/"configurar"(debería estar predefinida) la automatización.
 - c. Permitir consultar dispositivos.
3. Si el usuario admin inició sesión satisfactoriamente, permitir las siguientes acciones mediante un menú:
 - a. Permitir consultar automatizaciones activas
 - b. Gestionar dispositivos.
 - c. Permitir modificar el rol de un usuario.

**** El usuario Admin, puede ser el primer usuario que se registra.****

Base de datos

1. **Modelo Relacional:**
 - a. En base a las devoluciones realizadas sobre el DER, hacer las correcciones que sean necesarias.
 - b. Establecer las relaciones entre las entidades.
 - c. Crear el modelo relacional en base al diseño planteado aplicando todas las FN.

Ética y Deontología Profesional

2. Informe de Impacto Tecnológico:

- a. Analizar el impacto social y económico del sistema SmartHome:
 - i. Beneficios para usuarios (ahorro energético, seguridad, conveniencia)
 - ii. Consideraciones sobre brecha digital y accesibilidad
 - iii. Impacto ambiental (eficiencia energética, sostenibilidad)
- b. Relacionar el análisis con el pilar de sostenibilidad del AWS Well-Architected Framework.

Entregables

- **Programación I.** Link a la organización en Github + link al repositorio remoto (sólo se evaluará la rama main).

*Todos los integrantes deben tener confirmaciones (commits) con líneas de código de valor. **Sólo se evaluarfi la rama main** (entregable a producción)*

*Nota: Se **valorarfi** modularidad, código que respeta las nomenclaturas estándar de la comunidad de Python. Cumplimiento de requerimientos no funcionales (modularidad, usabilidad y legibilidad). En relación a las*

confirmaciones (commits), se valorará que éstas sean atómicas e incluyan mensajes descriptivos.

- **Base de Datos.** Documento que contenga el DER entregado anteriormente, describa de manera concisa las correcciones hechas y el modelo relacional resultante de ese diagrama. El mismo lo pueden hacer con la herramienta <https://app.diagrams.net/>, <https://www.lucidchart.com/> o cualquiera que ustedes definan.
- **Ética y Deontología Profesional.**
 - a. Informe completo de ética y deontología aplicada al proyecto (4-8 páginas) que integre todos los documentos anteriores y añada el análisis de impacto y el plan de actualización.
 - b. Manual ético para usuarios finales (2-3 páginas).

Entregables

1. Programación I

- Link a repositorio de github (publico): <https://github.com/LuisAsensio1/ABP.git>
- Video Demo: se anexara en la entrega de la evidencia 03 como archivo mp4

Justificación de la Automatización Elegida

SmartHome - Cámaras de Seguridad

Elegimos automatización por horarios

Cuando empezamos a pensar en qué tipo de automatización implementar para los dispositivos, teníamos varias opciones: detección de movimiento, detección de luz, o por horario. Al final nos decidimos por la automatización programación horaria porque nos pareció la más práctica para implementar y la que más sentido tiene para un usuario real, que busca lograr un equilibrio entre usabilidad, eficiencia energética y comodidad.

Pensamos en casos concretos: si tenés un local comercial, probablemente querés que los dispositivos del domicilio como, por ejemplo cámaras de un local comercial, graben durante el horario laboral (por ejemplo de 8 a 20hs) y que apaguen la grabación durante la noche, para no llenar la memoria con grabaciones de un comercio vacío. O al revés, si es para el domicilio de una casa, capaz querés que grabe de noche cuando no hay nadie despierto y estás vulnerable, pero no durante el día.

La detección de movimiento era otra opción posible, pero era más complicada de programar y no teníamos tanto tiempo. Además, con los horarios es más predecible: vos sabés exactamente cuándo el dispositivo va a ejecutar una acción o no.

¿Cómo funciona el sistema?

A continuación desarrollamos su explicación:

1. Diccionarios utilizados:

El sistema almacena los datos de nuestros usuarios, sus dispositivos y automatizaciones configuradas en estructuras de datos. Para nuestro sistema elegimos almacenar datos en diccionarios, donde cada uno se relaciona con el otro a través de sus claves.

```
usuarios[email] = {
```

```
    "nombre": nombre,
```

```
    "contrasena": contrasena,
```

```
    "rol": rol
```

```
}
```

```
dispositivos[email][nombre] = {
```

```
    "tipo": tipo_disp,
```

```
    "modelo": modelo,
```

```
    "estado_disp": False
```

```
}
```

```

automatizaciones[email_actual][dispositivo]["programacion_horaria"] = {
    "estado": nuevo_estado,
    "hora_encendido": hora_encendido,
    "hora_apagado": hora_apagado
}

```

Como podemos ver, en este último las claves hora_encendido y hora_apagado guardan los horarios que el usuario configura. Y la clave estado almacena si está activa o no.

2. La función para configurar los horarios que utilizamos fue:

```

def configurar_automatizacion(email_actual, dispositivo, nuevo_estado, hora_encendido,
hora_apagado):

```

```

    if email_actual not in automatizaciones:

```

```

        automatizaciones[email_actual] = {}

```

```

    if dispositivo not in automatizaciones[email_actual]:

```

```

        automatizaciones[email_actual][dispositivo] = {}

```

```

    automatizaciones[email_actual][dispositivo]["programacion_horaria"] = {
        "estado": nuevo_estado,
        "hora_encendido": hora_encendido,
        "hora_apagado": hora_apagado
    }

```

```

    print(f"✅ La automatización para el '{dispositivo}' fue configurada correctamente.")

```

Lo que primero hace es validar que exista una automatización registrada en el diccionario con el email del usuario logueado, dicho email es la clave que almacena en su valor sus automatizaciones, de encontrarse vacía, inicializa un diccionario vacío para la automatización. De la misma manera, lo hace con los dispositivos. Luego toma los

parámetros que el mismo usuario ingresa a través del menú con los diferentes inputs y almacena la información en los valores de las diferentes claves (nuevo_estado, hora_encendido, hora_apagado).

3. La lógica de la siguiente función decide si ejecuta la acción o no:

```
def ejecutar_automatizacion(email_actual):
```

```
    if email_actual not in automatizaciones:
```

```
        print("❌ Usuario no encontrado.")
```

```
        return
```

```
    # Conseguir la hora actual con datetime.now() y darle formato str con .strftime("%H:%M")
```

```
    ahora = datetime.now().strftime("%H:%M")
```

```
    for dispositivo, datos in automatizaciones[email_actual].items():
```

```
        programado = datos.get("programacion_horaria")
```

```
        if not programado or not programado.get("estado"):
```

```
            continue # De no encontrar una automatización activa, saltamos con continue
```

```
    # Verificar si la hora actual está dentro del rango de encendido
```

```
    hora_encendido = programado["hora_encendido"]
```

```
    hora_apagado = programado["hora_apagado"]
```

```
    # Validación considerando cruce de medianoche
```

```
    if hora_encendido <= hora_apagado: # misma jornada
```

```
        encendido = hora_encendido <= ahora <= hora_apagado
```

```
    else: # cruza medianoche
```

```
        encendido = ahora >= hora_encendido or ahora <= hora_apagado
```



```

datos["estado_disp"] = encendido

if email_actual in dispositivos and dispositivo in dispositivos[email_actual]:

    dispositivos[email_actual][dispositivo]["estado_disp"] = encendido

estado_texto = "encendido" if encendido else "apagado"

print(f"✅ Su dispositivo: {dispositivo} se encuentra {estado_texto} automáticamente a las {ahora}.")

```

La función ejecutar_automatizacion() se encarga de comparar la hora actual con la hora configurada para determinar si la automatización cambia el estado de los dispositivos y si estos se encuentran ejecutando una acción. Si la hora actual se encuentra en el rango de horarios configurados, el o los dispositivos del hogar ejecutan la acción.

Validaciones implementadas

Una cosa importante fue validar que el usuario ingrese los horarios en un formato correcto. Hicimos una función para pedirle la hora al usuario:

```

def pedir_hora(mensaje):

    while True:

        hora_str = input(mensaje)

        try:

            hora_obj = datetime.strptime(hora_str, "%H:%M")

            return hora_obj.strftime("%H:%M")

        except:

            print("Debe ingresar un horario valido. HH:MM")

```

Si el usuario ingresa incorrectamente el horario como "25:99" o "a la noche", el programa le pide que ingrese de nuevo los datos hasta que lo haga bien. El formato que usamos es HH:MM y sistema de horario de 24 horas, que es el más estándar.

Ventajas de la automatización elegida

1. Ahorro energético:: Si el dispositivo ejecuta su acción durante 8-12 horas por día en lugar de las 24 horas, el gasto energético disminuye notablemente.
2. Más fácil de usar: El usuario solo tiene que configurar una vez los horarios y después el dispositivo maneja sus funciones automáticamente. Así mismo, el usuario no tiene que usar el dispositivo manualmente, aumenta la comodidad y el confort.
3. Flexibilidad: Funciona para cualquier caso de uso, si tienes un comercio o si es para tu casa, entre otros usos.
4. Previsibilidad: A diferencia de la detección de movimiento esta automatización no depende de un evento externo, más que solo guiarse por la hora actual y los horarios configurados.

Posibles mejoras a futuro

1. Configurar diferentes horarios para diferentes días de la semana (por ejemplo: horarios de lunes a viernes y otros horarios para el fin de semana).
2. Enviar notificaciones reales cuando empieza/termina la acción.
3. Permitir configurar múltiples rangos horarios en el mismo día (por ejemplo: de 08-12 y de 14-18).
4. Guardar en una base de datos cuándo se activó y cuándo se desactivó la grabación para luego analizar esos datos.

Conclusión:

La automatización horaria nos pareció la mejor opción para este proyecto porque es práctica, útil en la vida real, y estaba dentro de nuestras capacidades de programación. Nos llevó tiempo entender la lógica en general, pero al final quedó funcionando bien. El sistema hace exactamente lo que necesita hacer: ejecutar la acción en los horarios configurados y dejar de hacerlo fuera de ellos, todo de forma automática, con una sola intervención manual del usuario (cuando la configura).

2. Base de datos.

- Se entrega DER con las correcciones implementadas y eliminadas las entidades y atributos que teníamos planteadas a futuro, por las que solo existen de momento. Por ejemplo, aunque vemos coherente la existencia de cuenta, en esta evidencia 03 aun no contamos con dicha entidad y atributos.

Link de DER corregido:

<https://drive.google.com/file/d/1MmR5N19nZXfPa-y5IzRgallUY19rTvz2/view?usp=sharing>

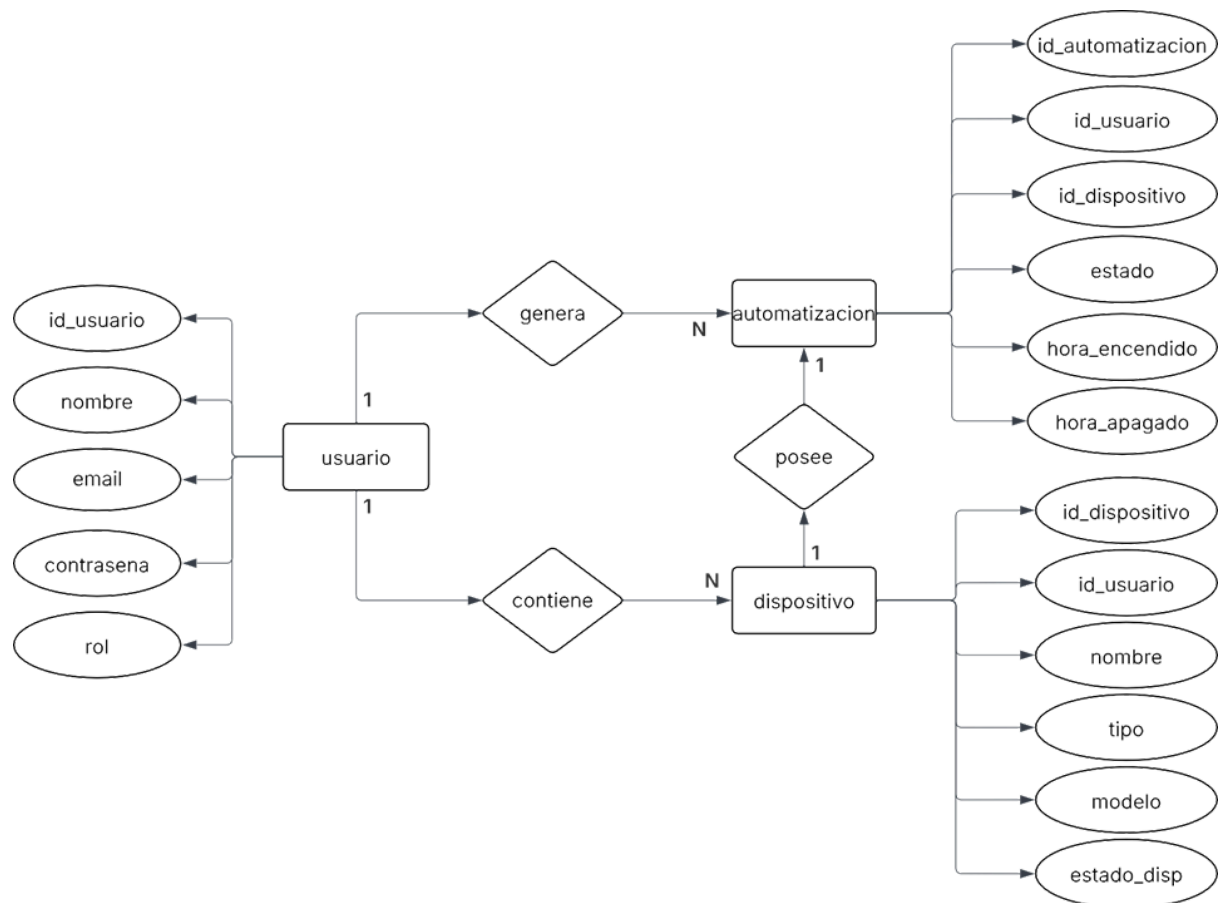
- Relaciones clave:
 - usuario 1:N dispositivo
 - usuario 1:N automatizacion
 - dispositivo 1:1 automatizacion
- Cardinalidad:

Relaciones Entre Entidades			
Entidad origen	Entidad destino	Tipo de relación	Clave foránea
usuario	dispositivo	1 a N	id_usuario
usuario	automatizacion	1 a N	id_usuario
dispositivo	automatizacion	1 a 1	id_dispositivo

Diagrama Entidad Relación – Corregido

Link al diagrama:

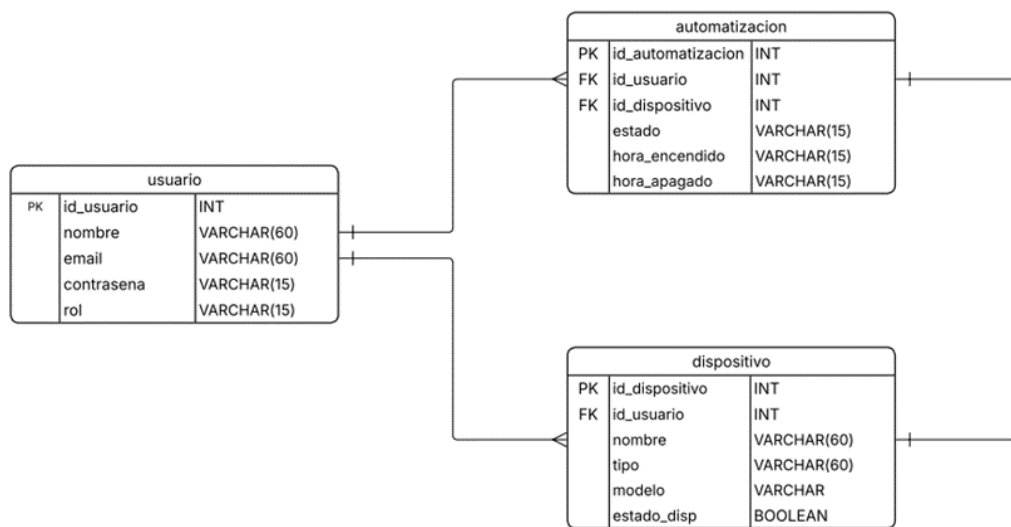
https://lucid.app/lucidchart/f1fb8b36-2c2e-4e23-8d12-a18ba90c546e/edit?beaconFlowId=74CC2B5C07CE14B7&invitationId=inv_a2160fca-03a5-4d67-ad95-66f5d68e9424&page=0_0#



Modelo Relacional - Corregido

Link al Modelo relacional:

https://lucid.app/lucidchart/5de8b248-9531-4a03-8ce9-f9735b3e5149/edit?beaconFlowId=971A42CC0050E159&page=0_0&invitationId=inv_a69ce4cb-08c5-4451-ab07-50971b0980cc#



3. Ética y Deontología Profesional

Política de Privacidad y Protección de Datos del Sistema SmartHome

1. Introducción

SmartHome Solutions reconoce la importancia de proteger la privacidad de sus usuarios y se compromete a manejar los datos personales de manera ética, responsable y conforme con la legislación vigente. Esta política se basa en los principios establecidos por la **Ley 25.326 de Protección de Datos Personales de Argentina** y el **Reglamento General de Protección de Datos (GDPR)** de la Unión Europea.

2. Tipos de datos personales que se recolectarán

El sistema recopilará información que permita identificar directa o indirectamente a una persona:

- **Datos identificatorios:** nombre, correo electrónico, teléfono.
- **Datos de ubicación:** dirección del domicilio y geolocalización de dispositivos.
 - **Datos técnicos y de uso:** registros de actividad en el sistema, configuraciones, automatizaciones.
- **Identificadores en línea:** direcciones IP, cookies técnicas (cuando aplique).

No se recolectarán datos personales sensibles (como salud, orientación sexual, religión, etc.), salvo que el usuario los introduzca voluntariamente, en cuyo caso se garantizará su protección reforzada conforme al artículo 7 del GDPR y el artículo 7 de la Ley 25.326

3. Finalidad del tratamiento de datos

El tratamiento de los datos personales tendrá fines **determinados, explícitos y legítimos**:

- Verificar la identidad del usuario (registro e inicio de sesión).

- Gestionar dispositivos y automatizaciones personalizadas.
- Enviar notificaciones sobre el estado del hogar inteligente.
- Mejorar el rendimiento del sistema a través de análisis de uso.
- Cumplir con requisitos legales y de seguridad.

Los datos **no serán utilizados para fines incompatibles** con los mencionados, como publicidad no autorizada, sin consentimiento expreso del usuario.

4. Medidas de seguridad implementadas

Siguiendo el principio de *integridad y confidencialidad* del GDPR y la Ley 25.326, se aplicarán las siguientes medidas:

- **Cifrado de datos** en tránsito y reposo.
 - **Control de acceso y contraseñas robustas.**
- **Autenticación multifactor.**
- **Gestión de roles y permisos** para evitar accesos no autorizados.
 - **Evaluaciones periódicas de riesgo (DPIA)**, conforme al artículo 35 del GDPR.

5. Derechos de los usuarios sobre sus datos

De acuerdo con el GDPR y la Ley 25.326, el usuario (titular de los datos) posee los siguientes derechos:

- **Derecho de información:** conocer quién trata sus datos y con qué fines.
- **Derecho de acceso:** obtener copia de sus datos procesados.
- **Derecho de rectificación:** corregir errores o datos desactualizados.
 - **Derecho de supresión ("derecho al olvido"):** solicitar la eliminación de sus datos cuando ya no sean necesarios o si retira su consentimiento.
 - **Derecho de oposición:** rechazar ciertos tratamientos (por ejemplo, publicidad).
- **Derecho a la portabilidad:** recibir sus datos en un formato estructurado.

- **Derecho a la limitación del tratamiento:** restringir temporalmente su uso.

Estos derechos pueden ejercerse mediante solicitud escrita al responsable de datos de SmartHome Solutions. Se garantizará una respuesta clara y dentro de los plazos establecidos por ley.

6. Base legal y cumplimiento normativo

Esta política se fundamenta en:

- **Ley 25.326 de Protección de Datos Personales (Argentina):** reconoce los *derechos ARCO+* (Acceso, Rectificación, Cancelación, Oposición e Información).
- **Reglamento General de Protección de Datos (GDPR):** aplicable como estándar internacional en caso de usuarios en la Unión Europea.
- **Habeas Data** (art. 43 de la Constitución Argentina): como garantía judicial para exigir acceso, corrección o eliminación de datos.

7. Notificación de violaciones y responsabilidades

En caso de violación de seguridad, el responsable del tratamiento deberá notificar a la autoridad competente **dentro de las 72 horas**, según el artículo 33 del GDPR. También se informará al usuario si hay riesgo significativo para sus derechos y libertades

8. Datos de contacto

El usuario podrá contactar al servicio de soporte técnico, servicio de datos personales a los siguientes correos electrónicos y pagina web:

www.smarthome.com

Soporte@smarthome.com

administracion@smarthome.com

Análisis del Impacto Social y Económico del Sistema SmartHome

Beneficios para los usuarios Ahorro

energético

- El sistema permite automatizar dispositivos y activar modos como la **“Programación horaria”**, lo que reduce el consumo eléctrico al apagar dispositivos en rangos de horarios que no son esenciales.

Seguridad

- A través de dispositivos como cámaras, sensores y notificaciones, el sistema fortalece el accionar de los mismos en horarios donde es fundamental hacer foco en monitorear el hogar o el local ante posibles eventos.

Conveniencia

- La posibilidad de controlar luces, electrodomésticos y climatización desde una sola interfaz centralizada mejora significativamente la **experiencia del usuario**.
- Las automatizaciones personalizadas (como encender la cafetera a las 7:00 am o activar el "Modo Noche") agregan confort y eficiencia a la rutina diaria.

Consideraciones sobre brecha digital y accesibilidad Brecha

digital

- El sistema requiere conexión a internet, dispositivos inteligentes y conocimientos básicos de tecnología, lo cual **excluye a sectores vulnerables** (adultos mayores, personas con escasos recursos o zonas rurales sin conectividad).

- La necesidad de smartphone u otros dispositivos de acceso puede **profundizar desigualdades tecnológicas** si no se acompaña con políticas inclusivas.

Accesibilidad

- No se detallan características específicas de accesibilidad (por ejemplo: soporte para usuarios con discapacidad visual o motriz), por lo que **debe incorporarse diseño universal o accesible**.
- Es fundamental aplicar principios de usabilidad y accesibilidad digital (WCAG, buenas prácticas de UX/UI) para que el sistema sea realmente inclusivo.

Impacto ambiental

Eficiencia

energética

- Al permitir el apagado automatizado de dispositivos y el ajuste dinámico del consumo, SmartHome promueve una **reducción de la huella energética del hogar**.
- La programación eficiente del uso de dispositivos evita el desperdicio y fomenta una cultura de consumo responsable.

Sostenibilidad

- El proyecto se alinea con el **pilar de Sostenibilidad del AWS Well-Architected Framework**, promoviendo prácticas responsables tanto a nivel de infraestructura (uso eficiente de recursos en la nube) como en el impacto final del sistema en los hogares.
 - Además, fomenta la **conciencia ambiental del usuario**, al visibilizar su consumo y permitir decisiones informadas.

Conclusión

El sistema **SmartHome** tiene un **impacto positivo** potencial en cuanto a **eficiencia, seguridad y confort del usuario**, así como en la **reducción del consumo energético doméstico**. Sin embargo, para maximizar sus beneficios sociales, debe

abordar activamente la brecha digital y los desafíos de accesibilidad. Integrar estos aspectos desde el diseño permitirá no solo cumplir con los pilares éticos y técnicos del proyecto, sino también aportar a una tecnología más equitativa y sostenible.

Manual de Usuario Final – SmartHome Solutions

INDICE

Introducción	1
SmartHome-----	1
Requisitos del Sistema-----	1
Instalación del Programa	1
Descarga e Instalación	1
Crear Cuenta o Iniciar Sesión-----	1
Configuración	2
Administrador-----	2
Menú Administrador	2
Administrar Dispositivos-----	2
Usuario-----	2
Uso básico del sistema	3
Soporte Técnico	3

Introducción

SmartHome Solutions

SmartHome Solutions es un sistema inteligente que permite controlar los dispositivos inteligentes conectados en el hogar. A través de nuestra plataforma el usuario puede gestionar cámaras, sensores de movimiento, luces y otros dispositivos compatibles desde una sola aplicación que es compatible con celulares, tablets y computadoras.

El objetivo principal de nuestra app, es brindar confort, seguridad y eficiencia energética al usuario, permitiendo monitorear los estados de cada dispositivo, activar sus automatizaciones y personalizar su funcionamiento adecuándose a las preferencias y horarios.

Requisitos del Sistema

Los requisitos del sistema para asegurar el correcto funcionamiento de nuestro programa son:

- Sistemas Operativos Compatibles: Windows, macOS, Linux, Android, iOS.
- Conexión a Wi-Fi estable.
- Dispositivos compatibles: (cámaras de seguridad, sensores de movimiento, luces inteligentes, etc).

Instalación del Programa

Descarga e Instalación

Para poder usar nuestro programa, usted debe descargar el instalador o aplicación de acuerdo al dispositivo que utilice:

- Para computadoras (Windows, MacOS, Linux):

Ingresa a nuestra página web oficial www.smarthomesolutions.com y haga click en la opción “Descargar para Windows” o el sistema operativo que utilice su dispositivo. Una vez que finalice la descarga ejecutar el archivo instalador “SmartHomeSetup.exe” y acepte los permisos requeridos.

- Para dispositivos móviles:

Android: ingrese a Google Play Store y busque “SmartHome Solution” y presione instalar la app.

iOS: ingrese a App Store y busque “SmartHome Solution” y presione obtener para instalar la app.

Crear Cuenta o Iniciar Sesión

Si es la primera vez que ingresa a nuestro sistema y no está registrado, usted primero debe crearse una cuenta siguiendo los siguientes pasos:

1. Seleccione la opción “Registrar Usuario”.
2. Completar los campos requeridos:
 - Nombre de Usuario
 - Email
 - Contraseña: la misma debe contener al menos 6 caracteres, letras mayúscula y números

Si usted ya posee una cuenta, puede iniciar sesión de la siguiente manera:

1. Ingrese los campos requeridos:

- Email

○

Contraseña

Configuración

Administrador

Menú Administrador

Si usted posee la categoría de administrador, a través del menú administrador puede:

1. Ver automatizaciones activas: en esta opción puede ver todas las automatizaciones que se encuentran con estado activa.
2. Gestionar dispositivos: si selecciona esta opción, usted puede agregar, mostrar, eliminar o modificar un dispositivo.
3. Modificar rol de usuario: puede gestionar el rol del usuario especificado, y cambiarlo a usuario administrador o usuario.

Administrar Dispositiv

Si usted posee la categoría de administrador, puede gestionar los dispositivos inteligentes del hogar con las siguientes opciones en el menú gestión de dispositivos:

1. Agregar Dispositivo
2. Eliminar dispositivo: permite eliminar un dispositivo específico. Para esto, debe ingresar el campo requerido "email de usuario" y "Nombre del dispositivo".
3. Modificar Dispositivo: permite cambiar la configuración del dispositivo.

Usuario

Si usted es un usuario, puede realizar las siguientes opciones desde el menú usuario:

1. Ver mis datos: permite visualizar mis datos, como por ejemplo: email
2. Configurar automatización: permite modificar las automatizaciones de un dispositivo registrado del usuario. Como por ejemplo: Encendido/Apagado, Modo de grabación (siempre/movimiento), Programar horario automático, Notificaciones por movimiento, Detección de movimiento, Modo ahorro, Activación nocturna silenciosa.
3. Ver mis dispositivos: permite visualizar mis dispositivos registrados.

Usuario

Si usted es un usuario común, puede realizar las siguientes opciones desde el menú usuario:

1. Ver mis datos: permite visualizar mis datos, como por ejemplo: email
2. Configurar automatización: permite modificar las automatizaciones de un dispositivo registrado del usuario.
3. Ver mis dispositivos: permite visualizar mis dispositivos registrados.

Uso Básico del Sistema

Una vez iniciada la sesión en nuestro sistema, su dispositivo está listo para acceder y gestionar sus dispositivos o usuarios registrados (esto último dependiendo de la categoría de usuario que se posea).

De sus dispositivos inteligentes tendrá control sobre las siguientes opciones:

- Automatizar.

- Ejecutar automatización.

Su dispositivo será un control remoto, capaz de gestionar todos los dispositivos inteligentes de su hogar desde cualquier ubicación incluso fuera de casa, tan solo teniendo conexión a internet y usando nuestra aplicación móvil.

Soporte técnico

Si tiene inconveniente con el uso de SmartHome Solutions, usted puede contactarse con nosotros y nuestro equipo técnico por nuestros canales de comunicación y atención al cliente de la siguiente manera:

- **Correo electrónico:** soporte@smarthomesolutions.com
- **Teléfono:** 0800-555-5151
- **Sitio web:** www.smarthomesolutions.com/soporte
- **Horario de atención:** Lunes a viernes de 9:00 a 18:00 hs.
- **Acceder a nuestra sección de preguntas frecuentes (FAQ) en nuestro sitio web.**