

# Procedimiento de Instalación - Scape Room Software

---

## Tabla de Contenidos

- [Introducción](#)
  - [Requisitos Previos](#)
  - [Arquitectura de Red](#)
  - [Fase 1: Preparación del Hardware](#)
  - [Fase 2: Configuración de Red](#)
  - [Fase 3: Instalación del Software](#)
  - [Fase 4: Configuración de Dispositivos](#)
  - [Fase 5: Pruebas de Sistema](#)
- 

## Introducción

Este documento describe el procedimiento completo para instalar y configurar el sistema Scape Room en **4 salas idénticas**, todas conectadas a un Starlink como fuente de internet principal.

### Características del Despliegue

- **4 salas idénticas** con configuración casi igual
  - **Starlink** como proveedor de internet principal
  - **1 router por sala** (segmento: 192.168.18.1)
  - **IP fija del servidor:** 192.168.18.164 (en cada sala)
  - **Configuración basada en MAC address** para IP estática del servidor
- 

## Requisitos Previos

### Hardware Requerido por Sala

#### Servidor Principal

- **1 Mini PC** (Ubuntu 20.04+ o Windows 10+)
- Mínimo 8GB RAM, 4 cores CPU
- Disco duro con mínimo 100GB libres
- Tarjeta de red Ethernet (para obtener MAC address)
- Node.js 18+ y pnpm instalados

#### Arduinos (3 unidades por sala)

- **1 Arduino** para juego de botones
- **1 Arduino** para tablero de conexiones
- **1 Arduino** para lector NFC/RFID
- **1 Arduino** para Pelotas

## Dispositivos Cliente

- **1 iPad** - Admin iPad
- **1 Tablet Android/iPad** - Tablet Feedback
- **1 Mini PC** - Servidor Central (Ubuntu 20.04+ o Windows 10+)
- **1 Mini PC** - Buttons Game
- **1 Totem Táctil** - con SO Windows incluido
- **1 Laptop** - AI App
- **1 Monitor/Proyector** - Main Screen (conectado a una de las Mini PCs o al servidor)

## Red

- **1 Router** por sala (debe soportar reserva DHCP por MAC)
- Cables Ethernet (mínimo 4)
- Switch de red de 8 puertos (recomendado)

## Periféricos

- **1 Impresora Epson PM-525** (o compatible) conectada por WiFi
- Certificados SSL (ya incluidos en [/cert](#))

## Software Requerido

### En el Servidor

```
# Versiones mínimas
Node.js: v18.0.0+
pnpm: 9.0.0+
Git: 2.30+

# Para Linux
CUPS (sistema de impresión)

# Para Windows
PowerShell 5.1+
Controladores de impresora Epson
```

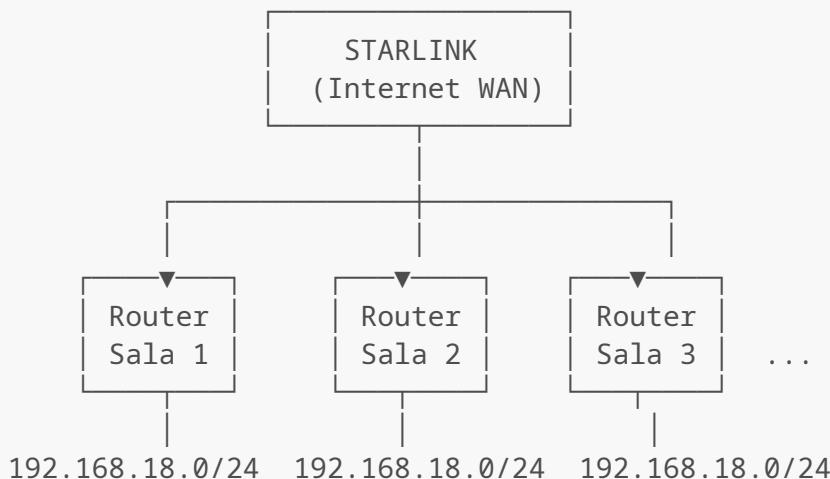
### En Dispositivos Cliente

- Navegadores web modernos:
  - Chrome 90+
  - Safari 14+
  - Firefox 88+
  - Edge 90+

---

## Arquitectura de Red

## Topología General (4 Salas)



## Segmento de Red por Sala

**CRÍTICO:** Todas las salas usan el mismo segmento de red.

Segmento de Red:	192.168.18.0/24
Gateway (Router):	192.168.18.1
Máscara de Subred:	255.255.255.0
Servidor:	192.168.18.164 (IP FIJA por MAC)
Impresora:	192.168.18.100-110 (DHCP)
Arduinos:	
- Botones:	192.168.18.101 (IP FIJA por MAC)
- Conexiones:	192.168.18.102 (IP FIJA por MAC)
- NFC:	192.168.18.103 (IP FIJA por MAC)
Clientes (DHCP):	192.168.18.XX
- Admin iPad:	192.168.18.XX
- Tablet Feedback:	192.168.18.XX
- Mini PC Buttons:	192.168.18.XX
- Totem Táctil:	192.168.18.XX
- Laptop AI App:	192.168.18.XX
- Main Screen:	192.168.18.XX

**IMPORTANTE:** Cada sala es una red aislada. No se comunican entre sí, solo comparten la conexión a internet del Starlink.

## Fase 1: Preparación del Hardware

### 1.1. Obtener MAC Address del Servidor

En cada sala, necesitas obtener la MAC address de la tarjeta de red del servidor para configurar la IP fija.

## Linux

```
# Obtener MAC address de la interfaz principal  
ip link show  
  
# O más específico (eth0 o wlan0)  
ip link show eth0 | grep link/ether  
  
# Ejemplo de salida:  
# link/ether aa:bb:cc:dd:ee:ff brd ff:ff:ff:ff:ff:ff  
# MAC Address: aa:bb:cc:dd:ee:ff
```

## Windows

```
# PowerShell  
Get-NetAdapter | Select-Object Name, MacAddress  
  
# CMD  
ipconfig /all  
  
# Busca "Dirección física" o "Physical Address"  
# Ejemplo: AA-BB-CC-DD-EE-FF
```

**ANOTA LA MAC ADDRESS DE CADA SERVIDOR** (una por sala):

```
Sala 1: aa:bb:cc:dd:ee:f1  
Sala 2: aa:bb:cc:dd:ee:f2  
Sala 3: aa:bb:cc:dd:ee:f3  
Sala 4: aa:bb:cc:dd:ee:f4
```

## 1.2. Obtener MAC Address de los Arduinos

Repite el proceso para cada Arduino. En el código Arduino, usa:

```
// Para WiFi  
WiFi.macAddress(mac);  
Serial.print("MAC Address: ");  
Serial.println(WiFi.macAddress());  
  
// Para Ethernet  
Ethernet.macAddress(mac);
```

Anota las MAC addresses:

Sala X - Arduino Botones: aa:bb:cc:dd:ee:a1  
Sala X - Arduino Conexiones: aa:bb:cc:dd:ee:a2  
Sala X - Arduino NFC: aa:bb:cc:dd:ee:a3

### 1.3. Verificar Funcionalidad de Hardware

- Servidor enciende correctamente
- Todos los Arduinos se conectan a la red
- Impresora está conectada al WiFi
- Todos los dispositivos cliente encienden
- Router está funcionando correctamente
- Cable Ethernet funciona en todos los puertos

---

## Fase 2: Configuración de Red

### 2.1. Configuración del Router

**IMPORTANTE:** Realiza estos pasos en el router de cada sala.

#### Acceder al Router

1. Conecta una PC al router vía Ethernet o WiFi
2. Abre navegador y ve a: <http://192.168.18.1>
3. Ingresa credenciales de administrador (usualmente admin/admin)

#### Configurar Segmento de Red

1. Navega a **Configuración de Red** o **LAN Settings**
2. Configura:

IP del Router: 192.168.18.1  
Máscara de Subred: 255.255.255.0  
DHCP Habilitado: Sí  
Rango DHCP: 192.168.18.20 - 192.168.18.200

#### Configurar Reservas DHCP (IP Fija por MAC)

**CRÍTICO:** Configura las siguientes reservas DHCP en el router.

Navega a **DHCP Reservation** o **Address Reservation**:

MAC Address del Servidor → 192.168.18.164  
MAC Arduino Botones → 192.168.18.101

MAC Arduino Conexiones	→ 192.168.18.102
MAC Arduino NFC	→ 192.168.18.103
MAC Arduino NFC	→ 192.168.18.104
MAC Impresora (opcional)	→ 192.168.18.105

### Ejemplo de configuración en router TP-Link:

- **Address Reservation** → Add New
- **MAC Address:** aa:bb:cc:dd:ee:ff
- **Reserved IP Address:** 192.168.18.164
- **Status:** Enabled
- Save

### Ejemplo en router Cisco:

- **LAN → DHCP Reservation**
- Agregar dispositivo manualmente
- Ingresar MAC y IP
- Apply

### Habilitar UPnP (opcional pero recomendado)

UPnP: Enabled

### Guardar y Reiniciar Router

1. Guarda todos los cambios
2. Reinicia el router
3. Espera 2-3 minutos para que se reinicie completamente

## 2.2. Verificar Conectividad del Servidor

### Conectar Servidor al Router

1. Conecta el servidor al router vía Ethernet
2. Espera 30 segundos para que obtenga IP
3. Verifica que tenga la IP correcta

### Linux

```
# Verificar IP asignada
ip addr show

# Deberías ver: 192.168.18.164
# Si no, reinicia la interfaz de red:
sudo systemctl restart NetworkManager
```

```
# 0 reinicia el servidor  
sudo reboot
```

## Windows

```
# PowerShell  
Get-NetIPAddress -AddressFamily IPv4  
  
# CMD  
ipconfig  
  
# Busca la interfaz de red activa  
# Deberías ver: 192.168.18.164  
  
# Si no, libera y renueva IP:  
ipconfig /release  
ipconfig /renew
```

## Pruebas de Conectividad

```
# Ping al router  
ping 192.168.18.1  
  
# Ping a internet  
ping 8.8.8.8  
  
# Ping al dominio  
ping samay.checkappia.com
```

**Si alguna prueba falla,** revisa:

- Configuración del router
- Reserva DHCP por MAC
- Cable Ethernet
- Firewall del servidor

## 2.3. Configurar Arduinos

### Verificar Conexión de Arduinos

1. Sube el código a cada Arduino
2. Abre el **Serial Monitor** (115200 baud)
3. Verifica logs de conexión

**Si falla la conexión:**

- Revisa reserva DHCP en router
  - Asegúrate de que el servidor esté encendido
  - Revisa firewall del servidor
- 

## Fase 3: Instalación del Software

### 3.1. Instalar Dependencias

```
# Instalar todas las dependencias
pnpm install
```

### 3.2. Verificar Certificados SSL

Los certificados ya están incluidos en [/cert](#):

```
ls -la cert/
# Deberías ver:
# privkey1.pem
# cert1.pem
# chain1.pem
# fullchain1.pem
```

**Si faltan los certificados**, contacta a Yeferson para obtenerlos.

### 3.3. Configurar Variables de Entorno del Servidor

Edita el archivo [.env](#):

```
# HTTP Server (para comunicación con Arduinos)
HTTP_PORT=3001

# HTTPS Server (para clientes web)
HTTPS_PORT=3443

# Host binding (IMPORTANTE: 0.0.0.0 para escuchar en todas las interfaces)
HOST=0.0.0.0

# Timeout de Arduino (30 segundos)
ARDUINO_TIMEOUT=30000

# Configuración de Impresora
# Linux: nombre de la impresora en CUPS (ejemplo: "Epson-PM-525")
# Windows: nombre exacto como aparece en "Impresoras y escáneres"
PRINTER_NAME=Epson-PM-525
```

```
# Ruta al PDF (opcional, por defecto usa: to-print/to-print.pdf)
#PDF_PATH=
```

### 3.4. Configurar Variables de Entorno de Clientes

**IMPORTANTE:** Todas las apps cliente usan el mismo archivo `.env`.

Para cada app en `apps/`:

- `admin-ipad`
- `ai-app`
- `buttons-game`
- `main-screen`
- `tablet-feedback`
- `totem-tactil`

Edita el archivo `.env`:

```
# URL del servidor WebSocket (MISMA EN TODAS LAS SALAS)
# IMPORTANTE: Usa la IP fija del servidor, NO el dominio
# Este es el puerto del servidor WebSocket (3443), no el de la app
VITE_SERVER_URL=https://192.168.18.164:3443
```

Configura el archivo `/etc/hosts` en cada dispositivo cliente:

```
# Linux/Mac
sudo nano /etc/hosts

# Windows (como administrador)
notepad C:\Windows\System32\drivers\etc\hosts

# Agregar:
192.168.18.164 samay.checkappia.com
```

### 3.5. Compilar el Proyecto

```
# Volver a la raíz del proyecto
cd ~/scape-room-software

# Compilar todas las apps y el servidor
pnpm run dev
```

**Si hay errores de compilación:**

- Revisa que todas las dependencias estén instaladas

- Ejecuta `pnpm install` de nuevo
- Revisa logs de error en terminal

## Fase 4: Configuración de Dispositivos

### 4.1. Configurar Impresora

#### Linux

```
# Instalar CUPS
sudo apt-get update
sudo apt-get install cups cups-client

# Iniciar servicio CUPS
sudo systemctl start cups
sudo systemctl enable cups

# Abrir interfaz web de CUPS
# En el navegador: http://localhost:631

# Agregar impresora:
# 1. Administration → Add Printer
# 2. Selecciona Epson PM-525 de la lista
# 3. Si no aparece, usa: ipp://192.168.18.105:631/ipp/print
# 4. Asigna nombre: "Epson-PM-525"
# 5. Completa el asistente

# Verificar impresora
lpstat -p
# Deberías ver: printer Epson-PM-525 is idle

# Imprimir prueba
lp -d Epson-PM-525 ~/scape-room-software/apps/server/to-print/to-print.pdf
```

#### Windows

```
# Abrir Configuración → Dispositivos → Impresoras y escáneres
# Agregar impresora o escáner
# Seleccionar Epson PM-525
# Instalar controladores si es necesario

# Verificar impresora
Get-Printer | Select-Object Name

# Imprimir prueba
Start-Process -FilePath "C:\ruta\al\pdf" -Verb Print
```

Ver detalles completos en: [docs/PRINTER-SETUP.md](#)

## 4.2. Configurar Clientes en Dispositivos

**IMPORTANTE:** Cada aplicación cliente corre en su propio puerto HTTPS. Asegúrate de usar el puerto correcto para cada app.

### Admin iPad

1. Abre Safari en el iPad
2. Ve a: <https://192.168.18.164:5177>
3. Safari mostrará advertencia de certificado
4. Toca **Avanzado → Visitar este sitio web**
5. La app debería cargar correctamente
6. **Agrega a pantalla de inicio** para acceso rápido:
  - Toca el icono de compartir
  - **Agregar a pantalla de inicio**

### Tablet Feedback

1. Abre Chrome/Safari
2. Ve a: <https://samay.checkappia.com:5175>
3. Acepta certificado
4. Agrega a pantalla de inicio

### Main Screen (Proyector)

1. Conecta monitor/proyector a servidor o Mini PC
2. Abre navegador en modo pantalla completa (F11)
3. Ve a: <https://192.168.18.164:5173>
4. Acepta certificado

### Totem Táctil

1. Abre navegador en el Totem
2. Ve a: <https://192.168.18.164:5176>
3. Acepta certificado
4. Habilita modo pantalla completa

### Buttons Game (Mini PC)

1. Abre navegador en la Mini PC de Buttons
2. Ve a: <https://192.168.18.164:5174>
3. Acepta certificado
4. Habilita modo pantalla completa (F11)

### AI App (Laptop)

1. Abre navegador en la Laptop

2. Ve a: <https://192.168.18.164:5178>
3. Acepta certificado
4. Mantén en modo pantalla completa o maximizado

## RESUMEN DE PUERTOS:

```
Puerto 3001: HTTP Server (comunicación con Arduinos)
Puerto 3443: HTTPS Server (WebSocket)
Puerto 5173: Main Screen
Puerto 5174: Buttons Game
Puerto 5175: Tablet Feedback
Puerto 5176: Totem Táctil
Puerto 5177: Admin iPad
Puerto 5178: AI App
```

## Fase 5: Pruebas de Sistema

### 5.1. Iniciar Servidor y Apps Cliente

```
# Desde la raíz del proyecto, esto inicia el servidor y todas las apps
pnpm run dev
```

#### Si el servidor no inicia:

- Revisa que los puertos 3001 y 3443 no estén ocupados
- Verifica que los certificados SSL existan en [/cert](#)
- Revisa permisos de archivos
- Consulta logs en [apps/server/logs/](#)

#### Si una app cliente no inicia:

- Revisa que el puerto específico no esté ocupado
- Verifica el archivo [.env](#) de la app
- Verifica los certificados SSL en [/cert](#)

### 5.2. Verificar Conexión de Arduinos

Revisa logs del servidor. Deberías ver:

```
[INFO] POST /connect
[INFO] Arduino buttons registrado: 192.168.18.101:8080
[INFO] POST /heartbeat - Arduino: buttons
```

#### Si un Arduino no aparece:

- Reinicia el Arduino
- Revisa Serial Monitor del Arduino
- Verifica IP del Arduino con `ping 192.168.18.101`
- Revisa firewall del servidor

### 5.3. Verificar Conexión de Clientes

Abre cada app cliente en su dispositivo correspondiente.

**En los logs del servidor deberías ver:**

```
[INFO] WebSocket connection from: 192.168.18.20
[INFO] Client registered: admin-ipad
[INFO] WebSocket connection from: 192.168.18.21
[INFO] Client registered: tablet-feedback
...
```

**En la consola del navegador (F12) deberías ver:**

```
Socket connected
Registered as: admin-ipad
State received: { session: {...}, arduinos: {...} }
```

**Si un cliente no conecta:**

- Verifica URL en `.env`
- Revisa consola del navegador (F12)
- Acepta certificado SSL si aparece advertencia
- Verifica conectividad con `ping 192.168.18.164`

### 5.4. Prueba de Flujo Completo

**Test 1: Admin iPad**

1. Abre Admin iPad
2. Verifica que veas el estado de todos los módulos
3. Crea una nueva sesión
4. Inicia el temporizador
5. Verifica que el Main Screen muestre el tiempo

**Test 2: Juego de Botones**

1. En Admin iPad, inicia el módulo de botones
2. Presiona los botones físicos en el Arduino
3. Verifica que la app Buttons Game actualice en tiempo real
4. Completa la secuencia correcta
5. Verifica que aparezca el código de completado

**Test 3: Tablero de Conexiones → Totem**

1. Conecta cables en el Arduino de conexiones
2. Completa todas las conexiones correctas
3. Verifica que el Totem muestre el código

**Test 4: Tablet Feedback → Main Screen**

1. Navega por las pantallas en Tablet Feedback
2. Verifica que el Main Screen proyecte en tiempo real
3. Escribe feedback
4. Verifica que aparezca en Main Screen

**Test 5: AI App → Impresora**

1. Completa el flujo de AI App
2. Llega a la vista final
3. Verifica que el servidor envíe el PDF a la impresora
4. Confirma que la impresora imprime el documento

**5.5. Prueba de Resiliencia****Prueba de Heartbeat de Arduinos**

1. Desconecta un Arduino
2. Espera 40 segundos
3. Verifica que el servidor lo marque como desconectado
4. Reconecta el Arduino
5. Verifica que el servidor lo marque como conectado

---

**Resumen de IPs y Puertos por Sala****IPs (TODAS FIJAS POR MAC)**

Router:	192.168.18.1
Servidor:	192.168.18.164 (FIJA por MAC)

Arduinos (FIJAS por MAC):	
Botones:	192.168.18.101
Conexiones:	192.168.18.102
NFC:	192.168.18.103

Impresora:	192.168.18.105 (opcional fija)
------------	--------------------------------

Clientes (DHCP):	192.168.18.20-50
------------------	------------------

**Puertos del Servidor**

Puerto 3001: HTTP Server (comunicación con Arduinos)  
Puerto 3443: HTTPS Server (WebSocket para todas las apps)

## Puertos de Apps Cliente (HTTPS)

Puerto 5173: Main Screen	→ https://192.168.18.164:5173
Puerto 5174: Buttons Game	→ https://192.168.18.164:5174
Puerto 5175: Tablet Feedback	→ https://192.168.18.164:5175
Puerto 5176: Totem Táctil	→ https://192.168.18.164:5176
Puerto 5177: Admin iPad	→ https://192.168.18.164:5177
Puerto 5178: AI App	→ https://192.168.18.164:5178

### NOTA CRÍTICA:

- Todas las apps cliente se conectan al WebSocket del servidor en el puerto **3443**
- Cada app cliente sirve su interfaz web en su propio puerto (5173-5178)
- Los Arduinos se comunican con el servidor vía HTTP en el puerto **3001**

## Documentos Relacionados

- [Checklist de Instalación](#)
- [Configuración de Red](#)
- [Solución de Problemas](#)