

Manual de referencia: Make

Datos de esta manual

Fase	MAN — Manuales de referencia
Módulo	MAN — Biblioteca de consulta

Objetivo práctico

Este manual reúne, en un único documento de consulta, todo lo que necesitas saber sobre Make más allá de la lección práctica del módulo 3.3: los planes y el sistema de créditos, el vocabulario completo (scenarios, modules, connections, bundles, mapping, filtros), la situación honesta de accesibilidad de su editor, la seguridad de las conexiones, la API oficial, y el catálogo de errores más frecuentes con su diagnóstico.

Resultado que obtendrás

No hay un resultado práctico único que producir: este manual se consulta, no se «completa». El resultado esperado es que sepas, en cualquier momento del curso, dónde buscar la respuesta correcta sobre planes, seguridad, accesibilidad o errores de Make sin depender de la memoria ni de fuentes de terceros no verificadas.

Dónde encaja dentro del sistema final

Make ocupa la pieza AUTOMATIZACIÓN del sistema TELÉFONO → AGENTE → CEREBRO → VOZ → AUTOMATIZACIÓN → CALENDARIO: recibe, mediante un Webhook, la petición que genera el AGENTE (Vapi o Retell) cuando el CEREBRO decide que hace falta reservar, consultar o modificar una cita, y se encarga de traducir esa petición en una acción real sobre el CALENDARIO (Google Calendar o Cal.com). Es, junto con n8n, una de las dos opciones intercambiables para esa pieza; el resto del sistema no cambia según cuál elijas.

Conocimientos previos

- Automatización accesible con Make (módulo 3.3, lección práctica de este mismo curso).
- Webhooks: cómo viaja la información entre sistemas (módulo 3.1).

Herramientas necesarias

- Una cuenta de Make ya creada.
- Un navegador con NVDA activo.

Posible coste

Verificado contra la página oficial de precios de Make (make.com/en/pricing) el 30 de agosto de 2026. En 2026, Make renombró su unidad de facturación de «operaciones» a **créditos**: el criterio de consumo es el mismo (cada llamada a un módulo, como añadir una fila en una hoja o consultar un contacto, consume un crédito; algunas funciones avanzadas, como ciertas acciones de IA, pueden consumir más de uno).

Free: 0 €/mes, hasta 1.000 créditos al mes, editor visual completo, acceso al catálogo de aplicaciones, límite de 2 escenarios activos simultáneamente, e intervalo mínimo de programación de 15 minutos entre ejecuciones automáticas.

Core: desde 9 €/mes (facturación anual, con 10.000 créditos/mes en el escalón inicial), escenarios activos ilimitados, programación cada 1 minuto, y acceso a la API de Make.

Pro: desde 16 €/mes (facturación anual, 10.000 créditos/mes en el escalón inicial), añade ejecución prioritaria, variables personalizadas y búsqueda de logs a texto completo.

Teams: desde 29 €/mes (facturación anual, 10.000 créditos/mes en el escalón inicial), añade roles de equipo, plantillas compartidas y usuarios ilimitados.

Enterprise: precio a medida, con créditos personalizables, soporte 24/7 y funciones personalizadas.

La facturación anual resulta más económica que la mensual en todos los planes de pago (en torno a un 15% o más de ahorro, según la página oficial). Para las prácticas de este curso, el plan Free es suficiente: 1.000 créditos al mes cubren con holgura las pruebas de un único escenario de reservas.

Advertencia de seguridad

Make documenta oficialmente que su seguridad para las conexiones (equivalentes a las credenciales de n8n) se apoya en el principio de mínimo privilegio: al crear una conexión con una aplicación externa (por ejemplo, Google Sheets), concede sólo los permisos que ese escenario necesita, no más. Para los Webhooks de entrada, Make recomienda protegerlos con un secreto compartido (una cadena de

verificación incluida en la petición) y, si es posible, restringir qué direcciones IP pueden invocarlos. Los secretos y claves de API que un escenario necesite deben guardarse en almacenes de datos (data stores) o en un gestor externo de secretos, no escritos directamente y de forma visible dentro de los módulos. Make cuenta además con una página oficial de privacidad y cumplimiento de RGPD, en la que declara procesar datos personales de forma lícita, leal y transparente, recogiendo únicamente con una finalidad específica y explícita, y conservándolos sólo el tiempo necesario para esa finalidad. Nunca importes un Blueprint de origen desconocido sin revisar antes qué conexiones y qué acciones ejecuta cada módulo.

Modelo mental

Piensa de nuevo en el mensajero interno de una oficina: recibe un recado, lo lleva al departamento apropiado y regresa con una respuesta. En Make, ese recorrido completo se llama *scenario*, y cada parada del mensajero se llama *module*: son, conceptualmente, los mismos elementos que en n8n, con nombres distintos.

Explicación

Este apartado reúne el vocabulario y los conceptos de Make que no se cubren en detalle en la lección práctica del módulo 3.3.

Cada vez que un módulo falla dentro de un escenario, puedes conectarle una **ruta de gestión de errores** (error handler route), un camino alternativo que se activa sólo cuando ese módulo concreto falla. A esa ruta se le asigna una **directiva de error**: «Ignore» registra el error y continúa el escenario con el siguiente elemento; «Resume» continúa saltándose únicamente el elemento fallido; «Rollback» o «Commit» detienen el resto del escenario y, según los módulos implicados, pueden revertir cambios parciales ya realizados. Elegir la directiva correcta importa especialmente en un escenario de reservas: no es lo mismo que un fallo puntual se ignore silenciosamente a que detenga por completo el registro de la cita.

Más allá del editor visual, Make ofrece una **API REST oficial** (documentada en el Make Developer Hub, developers.make.com) que permite controlar la plataforma —escenarios, conexiones, ejecuciones— mediante peticiones HTTP autenticadas con un token de usuario, sin abrir la interfaz gráfica. Es importante saber que, según la documentación oficial, la mayoría de los endpoints de esta API están reservados a planes de pago y pueden no estar disponibles en el plan Free.

El plan Free limita el escenario a un máximo de **2 escenarios activos** simultáneamente y a un **intervalo mínimo de programación de 15 minutos** entre

ejecuciones automáticas (frente al intervalo de 1 minuto disponible desde el plan Core); para un Webhook de entrada activado por una llamada telefónica real, esto no afecta a la velocidad de respuesta (los Webhooks se disparan al instante, no por sondeo programado), pero sí limita cuántos escenarios distintos puedes tener activos a la vez en el plan gratuito.

Vocabulario nuevo

Crédito (Make)

Piénsalo así: El vale que se gasta cada vez que el mensajero hace una gestión concreta en un departamento.

Unidad de facturación de Make (antes llamada «operación»): cada llamada a un módulo consume, típicamente, un crédito.

Ejemplo: Un escenario con tres módulos que procesa una cita consume, como mínimo, tres créditos por ejecución.

Ruta de gestión de errores (error handler route)

Piénsalo así: El protocolo de emergencia que sigue el mensajero si un departamento concreto no puede atenderle.

Camino alternativo conectado a un módulo, que se activa únicamente cuando ese módulo falla.

Ejemplo: Una ruta de error en el módulo de Google Sheets puede enviar una notificación si no logra escribir la fila de la cita.

Directiva de error

Piénsalo así: La instrucción que decide si, tras un contratiempo, el mensajero sigue con el siguiente recado, repite el mismo, o cancela todo el recorrido.

Regla que determina qué hace el escenario tras ejecutarse una ruta de error: ignorar, reintentar, reanudar, confirmar (commit) o revertir (rollback).

Ejemplo: Una directiva «Rollback» en el módulo de creación del evento evita que quede una cita registrada a medias si el paso siguiente falla.

API de Make (Make Developer Hub)

Piénsalo así: El teléfono directo con el departamento de gestión de la oficina.

API REST oficial de Make que permite gestionar escenarios, conexiones y ejecuciones mediante peticiones HTTP autenticadas, reservada mayoritariamente a planes de pago.

Ejemplo: Con un token de la API de Make, puedes activar un escenario mediante una petición HTTP en lugar de abrir el editor visual.

Preparación

Este manual no requiere preparación técnica adicional: puedes consultarlo con tu cuenta de Make abierta o cerrada. Si vas a poner en práctica algo de lo que leas aquí, ten tu cuenta de Make iniciada y NVDA activo.

Procedimiento paso a paso

1. **Contexto:** Vas a comprobar cuántos créditos ha consumido tu cuenta este mes, para no sorprenderte con el límite del plan Free.

Acción de teclado: Recorre el menú principal de Make con **Tab** hasta «Billing» o el nombre de tu organización, y activa el apartado de uso o «Usage».

Respuesta esperada de NVDA: NVDA debería anunciar un texto similar a «Usage» o «Créditos usados», seguido de una cifra y el total disponible en tu plan.

Qué significa: Esta cifra te indica cuántos créditos de los 1.000 mensuales del plan Free (o del límite de tu plan) llevas consumidos.

Acción siguiente: Si te acercas al límite, revisa qué escenario consume más créditos desde el historial de ejecuciones de cada uno.

2. **Contexto:** Vas a añadir una ruta de gestión de errores a un módulo de tu escenario, para que un fallo no pase desapercibido.

Acción de teclado: Con el módulo seleccionado (Enter sobre su icono), busca la opción «Add error handler» recorriendo el menú contextual del módulo con **Tab**.

Respuesta esperada de NVDA: NVDA debería anunciar «Add error handler» o

el texto equivalente dentro del menú del módulo.

Qué significa: Esta ruta se dibuja como una rama alternativa que sale del módulo y sólo se activa si ese módulo falla.

Acción siguiente: Añade a esa ruta un módulo de notificación (por ejemplo, un correo) y elige la directiva de error apropiada («Ignore», «Rollback» u otra) según si el resto del escenario debe continuar o detenerse.

3. **Contexto:** Vas a revisar, en el historial de ejecuciones, exactamente qué bundle procesó cada módulo en una ejecución concreta.

Acción de teclado: Abre «History» desde el menú del escenario y pulsa Enter sobre la ejecución que quieras inspeccionar.

Respuesta esperada de NVDA: NVDA debería anunciar la lista de módulos ejecutados, con su estado (correcto o con error) y el número de bundles procesados por cada uno.

Qué significa: Si un módulo procesó 0 bundles cuando se esperaba al menos 1, el problema está en el filtro o en el módulo anterior, no en ese módulo en sí.

Acción siguiente: Contrasta el mensaje de error mostrado con el apartado «Errores frecuentes» de este manual.

Posibles diferencias de interfaz

Make usa nombres de menú («Add error handler», «Usage», «History») que pueden variar ligeramente de posición según si accedes desde la vista de un escenario concreto o desde el panel general de la organización. El principio estable es siempre el mismo: el uso de créditos se consulta a nivel de organización, el historial de ejecuciones a nivel de escenario, y la gestión de errores a nivel de módulo individual.

Problema de accesibilidad y alternativa

Situación honesta verificada el 30 de agosto de 2026: no se ha encontrado, en la documentación oficial de Make consultada, una declaración de accesibilidad ni evidencia verificable sobre el nivel de accesibilidad con teclado y lector de pantalla de su editor visual de escenarios. Al compartir el mismo paradigma de lienzo de posiciones libres que n8n —donde sí existen barreras documentadas oficialmente—, este manual no afirma que el editor de Make sea accesible sin haberlo podido comprobar, ni tampoco afirma lo contrario sin evidencia: se limita a decir, con honestidad, que no está confirmado.

Lo que sí es accesible, confirmado: el selector de archivos estándar de Windows usado para importar un Blueprint, los formularios de configuración de cada módulo (conexión, mapping de campos, filtros), el buscador de aplicaciones por nombre (sin necesidad de conocer identificadores técnicos internos), el historial de ejecuciones, y la API REST documentada oficialmente en el Make Developer Hub.

Grado de autonomía realista: alto para importar Blueprints, configurar conexiones en formularios de módulo, activar/desactivar escenarios, gestionar rutas de error y diagnosticar ejecuciones; no confirmado, en la fecha de esta redacción, para el diseño visual completo de un escenario complejo desde cero directamente en el lienzo.

Errores frecuentes y diagnóstico

El escenario consume más créditos de los esperados y se acerca al límite del plan Free antes de fin de mes.

Cómo localizarlo: Cada módulo del escenario consume un crédito por bundle procesado; un escenario con varios módulos en cadena, o que procesa varios bundles por ejecución (por ejemplo, un array con varias citas), multiplica el consumo.

Solución: Revisa el apartado «Usage» de tu organización para ver qué escenario consume más, y valora si algún módulo puede eliminarse, combinarse, o si el volumen real de tu negocio requiere pasar al plan Core.

Un módulo falla y el resto del escenario se detiene por completo, dejando una cita a medio registrar.

Cómo localizarlo: No hay una ruta de gestión de errores conectada a ese módulo, por lo que Make aplica el comportamiento por defecto ante un fallo (detener el escenario).

Solución: Añade una ruta de error a los módulos críticos del escenario, con la directiva apropiada (por ejemplo, «Rollback» para evitar registros parciales, o «Ignore» con notificación si el escenario puede continuar sin ese paso).

Una petición a la API de Make devuelve un error de permisos, aunque el token parece correcto.

Cómo localizarlo: Según la documentación oficial, la mayoría de los endpoints de la API de Make están reservados a planes de pago; el plan Free puede no tener acceso a ese endpoint concreto.

Solución: Confirma en developers.make.com/api-documentation qué plan mínimo requiere el endpoint que estás usando, y valora si necesitas pasar a un plan de pago para esa funcionalidad concreta.

Comprobación final

Sabes que puedes usar este manual con eficacia si, ante un fallo real de un escenario de Make, eres capaz de: localizar la ejecución fallida en «History», identificar el módulo exacto donde ocurrió el problema, comprobar si tenía una ruta de gestión de errores conectada, y contrastar el mensaje de error con el apartado «Errores frecuentes» de este manual.

Qué acabas de conseguir

Tienes una referencia completa de Make —planes y créditos, vocabulario, seguridad, accesibilidad y diagnóstico de errores— a la que volver en cualquier momento del curso sin depender de la memoria ni de fuentes no verificadas.

Ejercicio práctico

Elige uno de los tres errores frecuentes descritos en este manual y explica, en dos o tres frases, cómo lo diagnosticarías paso a paso usando exclusivamente el historial de ejecuciones («History») y NVDA, sin necesidad de pedir ayuda externa.

Resumen de lo imprescindible

- Make usa créditos (antes llamados operaciones) como unidad de facturación; el plan Free ofrece 1.000 créditos/mes, 2 escenarios activos y programación mínima cada 15 minutos.
- La seguridad de las conexiones se basa en el principio de mínimo privilegio; los Webhooks de entrada deben protegerse con un secreto compartido y, si es posible, restricción de IP.
- No existe declaración oficial sobre la accesibilidad del editor visual de Make: este manual lo dice explícitamente en lugar de asumir nada. La importación de Blueprints, los formularios de módulo y la API son las vías confirmadas como accesibles.
- Las rutas de gestión de errores, con su directiva correspondiente (Ignore, Resume, Rollback, Commit), determinan qué ocurre exactamente cuando un módulo falla.

Estoy preparado para continuar si puedo...

- Calcular si el plan Free de Make es suficiente para el volumen esperado de un proyecto concreto.
- Añadir una ruta de gestión de errores a un módulo y elegir la directiva apropiada.
- Diagnosticar un fallo de escenario usando el historial de ejecuciones con NVDA.
- Explicar honestamente qué se sabe y qué no se sabe sobre la accesibilidad del editor de Make.

Fuentes

- [Make Pricing — Make \(consultado el 2026-08-30\)](#)
- [Make.com GDPR | General Data Protection Regulation — Make \(consultado el 2026-08-30\)](#)
- [Make API documentation — Make Developer Hub \(consultado el 2026-08-30\)](#)
- [Getting started \(Make API\) — Make Developer Hub \(consultado el 2026-08-30\)](#)
- [Error handling | Custom Apps Documentation — Make Developer Hub \(consultado el 2026-08-30\)](#)
- [Scenarios > Blueprints — Make Developer Hub \(consultado el 2026-08-30\)](#)
- [Scenario blueprints — Make \(centro de ayuda oficial\) \(consultado el 2026-08-30\)](#)

Última actualización de este contenido: **2026-08-30**.