

Lógica Backend

Motor de disponibilidad, callouts, procesos y validaciones por evento.

- [Motor de Disponibilidad](#)
- [Callouts y Flujo de UI](#)
- [Procesos y Eventos](#)

Motor de Disponibilidad

`AppointmentAvailabilityEngine` concentra la lógica que decide si una cita o un slot es válido para un empleado, servicio y fecha determinados.

Responsabilidades actuales

- Cargar una configuración activa por organización con fallback a organización `0`.
- Leer duración, precio e intervalo del servicio.
- Aplicar overrides de precio y duración por empleado-servicio.
- Validar que el empleado pueda prestar el servicio.
- Validar días libres de la organización.
- Validar días libres del empleado.
- Validar cobertura del horario semanal del empleado.
- Detectar conflictos con citas existentes pendientes o aprobadas.
- Aplicar límites diarios por empleado cuando correspondan.
- Crear slots persistentes para la UI de iDempiere.

Regla de conflicto

Una cita se considera en conflicto cuando otra cita activa del mismo empleado se cruza con el rango nuevo y tiene estado pendiente o aprobado.

```
existing.StartDateTime < newEnd
AND existing.EndDateTime > newStart
AND existing.Status IN ('PE', 'AP')
AND existing.IsActive = 'Y'
```

Generación de slots

La generación actual es específica por empleado. El proceso no genera para todos los empleados ni autoasigna el recurso; parte de una selección explícita de empleado, servicio y fecha.

Nota: esta decisión hace que el flujo de UI sea más claro: primero se elige quién atiende, luego qué servicio presta, después se generan horarios disponibles.

Reglas actuales vs reglas esperadas

Regla	Estado actual	Comportamiento esperado
Empleado presta servicio	Implementado con <code>CDS_EmployeeService</code> .	Se mantiene como regla obligatoria.
Día libre de organización	Implementado por organización. Aunque <code>CDS_OrgDayOff</code> tiene <code>CDS_AppointmentConfig_ID</code> , el motor actual no filtra por configuración.	Debe filtrar por configuración cuando el servicio resuelva una configuración específica.
Día libre de empleado	Implementado con <code>CDS_EmployeeDayOff</code> .	Se mantiene como bloqueo específico del empleado.
Horario del empleado	Implementado con <code>CDS_EmployeeScheduleLine</code> .	Debe validarse después del horario macro de la configuración.
Horario macro de agenda	Pendiente. No existe aún una tabla equivalente a horario de configuración.	Debe definir los días y rangos generales permitidos, por encima del horario del empleado.
Anticipación mínima	Pendiente. Existe <code>MinAdvanceHours</code> , pero no se valida todavía.	Debe bloquear citas creadas con menos horas de anticipación que las configuradas.
Anticipación máxima	Pendiente. Existe <code>MaxAdvanceDays</code> , pero no se valida todavía.	Debe bloquear citas demasiado futuras.
Conflictos con citas existentes	Implementado para estados <code>Pending</code> y <code>Approved</code> .	Se mantiene como validación obligatoria.
Aprobación automática o forzada	Pendiente. Existen campos de configuración y servicio, pero no se aplican en el evento.	Debe resolverse en <code>BeforeNew</code> usando configuración y servicio.

Roadmap del motor

- Resolver la configuración desde el servicio seleccionado, no solo desde la organización.
- Agregar validación de horario macro de configuración antes del horario del empleado.
- Aplicar `MinAdvanceHours` y `MaxAdvanceDays` al generar slots y al guardar citas.
- Filtrar días libres de organización por configuración cuando exista relación.
- Aplicar reglas de aprobación en el evento antes de validar el estado final de la cita.

Callouts y Flujo de UI

`CalloutAppointment` ayuda al usuario a capturar una cita desde iDempiere sin reemplazar la validación final del backend.

Comportamiento de callouts

Campo	Comportamiento
<code>CDS_AppointmentService_ID</code>	Copia precio y duración efectivos. Si existe empleado seleccionado, usa override de <code>CDS_EmployeeService</code> . Limpia slot y horas, pero no limpia empleado.
<code>AppointmentDate</code>	Limpia slot, hora inicial y hora final. No limpia empleado ni servicio.
<code>CDS_AppointmentSlot_ID</code>	Copia empleado, hora inicial, hora final, precio y duración desde el slot seleccionado.

Flujo recomendado en ventana

1. Seleccionar el empleado que atenderá la cita.
2. Seleccionar el servicio filtrado para ese empleado.
3. Seleccionar la fecha de la cita.
4. Ejecutar el proceso de generación de slots.
5. Seleccionar el slot generado.
6. Guardar la cita.

Importante: si un usuario modifica manualmente horas, precio o duración, el evento de modelo vuelve a validar la consistencia antes de guardar.

Procesos y Eventos

El plugin mantiene pocos procesos activos y concentra las reglas críticas en eventos de modelo.

Procesos activos

Proceso	Uso
<code>GenerateAppointmentSlots</code>	Genera registros <code>CDS_AppointmentSlot</code> para el empleado, servicio y fecha seleccionados. Puede resolver valores desde la cita actual cuando se ejecuta desde la ventana.
<code>ReopenAppointment</code>	Reabre una cita procesada cambiándola a pendiente y limpiando <code>Processed</code> mediante un contexto controlado.

Proceso `GenerateAppointmentSlots`

Este proceso se usa para crear horarios disponibles persistentes que luego pueden consultarse desde la ventana iDempiere o desde la API leyendo la tabla `CDS_AppointmentSlot`.

Parámetro	Requerido	Uso
<code>AD_Org_ID</code>	Sí, directo o resuelto desde la cita	Organización para cargar configuración y validar disponibilidad.
<code>CDS_AppointmentEmployee_ID</code>	Sí	Empleado para el cual se generan slots.
<code>CDS_AppointmentService_ID</code>	Sí	Servicio que define duración, precio e intervalo del slot.
<code>AppointmentDate</code>	Sí	Fecha para la cual se generan horarios disponibles.
<code>CDS_Appointment_ID</code>	Opcional	Si se ejecuta desde una cita, permite resolver organización, empleado, servicio y fecha desde el registro actual.

Antes de generar nuevos slots, el proceso desactiva slots previos activos del usuario actual. Si hay una cita, desactiva los slots asociados a esa cita. Si no hay cita, desactiva slots del mismo usuario, empleado, servicio y fecha.

Uso recomendado: desde la ventana de cita se selecciona empleado, servicio y fecha; luego se ejecuta `GenerateAppointmentSlots`; finalmente se selecciona un registro de `CDS_AppointmentSlot`.

Proceso ReopenAppointment

Este proceso reabre una cita procesada para permitir correcciones administrativas controladas.

Parámetro	Requerido	Uso
<code>CDS_Appointment_ID</code>	Sí, directo o resuelto desde el registro actual	Cita procesada que se desea reabrir.

Si la cita ya está abierta, el proceso informa que no requiere reapertura. Si está procesada, activa un contexto temporal `#CDS_ReopenAppointment`, coloca `Processed=N` y cambia el estado a `Pending`.

Validación por evento actual

`AppointmentModelValidator` se ejecuta antes de crear o cambiar una cita. Para citas pendientes o aprobadas valida los datos mínimos y la disponibilidad real.

- Cliente requerido.
- Empleado requerido.
- Servicio requerido.
- Fecha de cita requerida.
- Hora inicial y final requeridas.
- Hora inicial menor que hora final.
- Fecha de cita igual al día de la hora inicial.
- Duración mayor que cero.
- Precio no negativo.
- Empleado habilitado para prestar el servicio.
- Slot consistente con empleado, servicio, horas, duración y precio.
- Sin bloqueo por día libre de organización o empleado.
- Horario cubierto por agenda del empleado.
- Sin conflicto con otra cita activa pendiente o aprobada.

Estados y procesamiento actual

- Si el estado viene vacío, se asigna `Draft`.
- Los estados finales marcan `Processed = true`.
- Los estados no finales limpian `Processed`.
- Una cita procesada no se puede modificar salvo mediante reapertura controlada.

Roadmap de validaciones en eventos

Las siguientes reglas deben agregarse al evento para que el comportamiento sea consistente desde ventanas iDempiere y desde REST/API.

Validación pendiente	Momento sugerido	Regla esperada
Resolver configuración desde servicio	<code>BeforeNew</code> y <code>BeforeChange</code>	El servicio debe indicar qué configuración de agenda aplica. La cita usa esa configuración para anticipación, horario macro y aprobación.
Servicio con una sola configuración	Al guardar servicio o relación configuración-servicio	Un servicio no debe estar asociado a más de una configuración activa.
Anticipación mínima	<code>BeforeNew</code> y generación de slots	Si <code>StartDateTime</code> está antes de ahora + <code>MinAdvanceHours</code> , la cita o slot debe bloquearse.
Anticipación máxima	<code>BeforeNew</code> y generación de slots	Si <code>StartDateTime</code> supera hoy + <code>MaxAdvanceDays</code> , la cita o slot debe bloquearse.
Horario macro de configuración	<code>BeforeNew</code> , <code>BeforeChange</code> y generación de slots	El horario de agenda general debe contener el rango solicitado antes de evaluar el horario del empleado.
Forzar aprobación	<code>BeforeNew</code>	Si la configuración tiene <code>ForceApproval=Y</code> , la cita debe quedar pendiente aunque el servicio permita aprobación automática.
Autoaprobar si el servicio lo permite	<code>BeforeNew</code>	Si <code>AutoApproveIfServiceAllows=Y</code> y el servicio no requiere aprobación, la cita puede quedar <code>Approved</code> .

Roadmap de workflows y correos

Además de las validaciones, falta definir workflows y plantillas de correo para notificar al cliente cuando cambie el estado de una cita.

Flujo pendiente	Disparo sugerido	Resultado esperado
Notificar cita aprobada	Cambio de estado a <code>Approved</code> .	Enviar correo al cliente con fecha, hora, servicio y empleado.
Notificar cita cancelada	Cambio de estado a <code>Cancelled</code> .	Enviar correo al cliente con el motivo de cancelación cuando exista.
Notificar reprogramación	Cambio de fecha, hora o empleado.	Enviar correo con el nuevo horario confirmado.
Recordatorio de cita	Proceso programado antes de <code>StartDateTime</code> .	Enviar recordatorio al cliente según configuración.

También falta crear las plantillas de correo, definir destinatarios, configurar remitente, parametrizar cuándo enviar cada notificación y evitar reenvíos duplicados con campos de control si aplica.

Orden recomendado de validación

1. Resolver servicio, empleado y organización.
2. Resolver configuración de agenda desde el servicio.
3. Aplicar reglas de anticipación mínima y máxima.
4. Validar horario macro de configuración.
5. Validar días libres de organización vinculados a la configuración.
6. Validar relación empleado-servicio.
7. Validar horario y días libres del empleado.
8. Validar slot seleccionado.
9. Validar conflictos con citas existentes.
10. Resolver estado de aprobación: pendiente o aprobada.