

SCRUM

EN ODOO 20

JUEGA EL PARTIDO COMPLETO



Marlon Falcón Hernández

IMPLEMENTO BIM 2.0 · GUÍA PRÁCTICA

1-5 JUGADORES

Sprints · Backlog · Burndown

Calentamiento: por qué Scrum y por qué Odoo 20

Scrum y Odoo 20 encajan bien porque Odoo ya tiene casi todas las piezas de la cancha: tareas, etapas Kanban, hitos, partes de horas, planificación y portal de cliente. Lo que falta es el reglamento, y eso es lo que aporta este libro.

Un equipo de baloncesto no gana por tener buenos jugadores, sino por jugar con un sistema. Scrum es ese sistema para proyectos: ciclos cortos, un objetivo claro por ciclo y una revisión honesta al final de cada uno. Odoo es el pabellón donde se juega: el marcador, el banquillo, la pizarra del entrenador y las estadísticas.

A quién va dirigido

- **Consultores e implantadores de Odoo** que quieren gestionar sus propios proyectos de implantación con Scrum.
- **Equipos de desarrollo** que programan módulos a medida y necesitan un backlog vivo.
- **Responsables de producto o de operaciones** que quieren ver el avance real sin pedir informes por correo.
- **Empresas de servicios** que facturan por horas o por hitos y necesitan que Scrum no rompa la facturación.

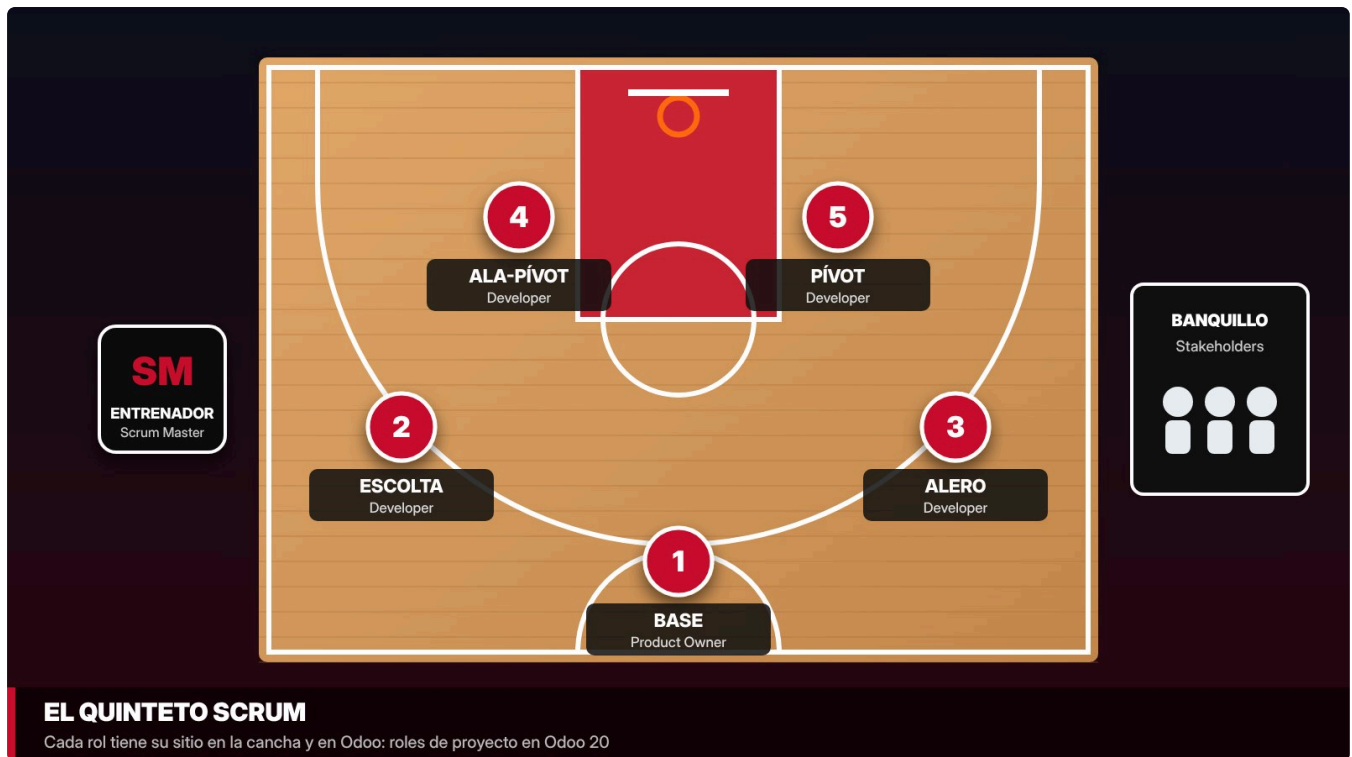
Cómo leer este libro

Cada capítulo es una fase del partido. Los capítulos 1 y 2 explican el reglamento y preparan la cancha. Del 3 al 7 se juega un Sprint completo, de la planificación a la retrospectiva. Los capítulos 8 y 9 añaden automatización y escala. El 10 es un partido real de principio a fin, y los apéndices traen la checklist y un módulo técnico de ejemplo.

Nota de versión. Odoo 20 se publicó en septiembre de 2026. Los nombres de menús pueden variar ligeramente según la edición (Community o Enterprise), el idioma y las apps instaladas. Cuando una función es exclusiva de Enterprise, se indica.

Capítulo 1 — El reglamento: Scrum en esencia

Scrum se resume en tres roles, cinco eventos y tres artefactos; todo lo demás es táctica. Antes de tocar Odoo conviene tener claro qué es cada pieza y dónde va a vivir dentro del sistema.



Los jugadores (roles)

- **Product Owner (base):** reparte el juego. Decide qué se construye y en qué orden, y es el único que ordena el Product Backlog.
- **Scrum Master (entrenador):** no juega, pero hace que el equipo juegue mejor. Protege el proceso, elimina bloqueos y facilita los eventos.
- **Developers (quinteto en pista):** construyen el incremento. Se autoorganizan y deciden cómo convertir el backlog en trabajo terminado.

Los tiempos del partido (eventos)

1. **Sprint:** el cuarto. Un periodo fijo de una a cuatro semanas; dos es lo más habitual.
2. **Sprint Planning:** el salto inicial. Se elige el objetivo del Sprint y qué elementos del backlog entran.
3. **Daily Scrum:** 15 minutos diarios para sincronizar y detectar bloqueos.
4. **Sprint Review:** se enseña el incremento a los interesados y se recoge su opinión.
5. **Sprint Retrospective:** el vestuario. El equipo revisa cómo ha jugado y elige qué mejorar.

El material (artefactos)

- **Product Backlog:** la lista ordenada de todo lo que podría construirse.
- **Sprint Backlog:** lo que el equipo se compromete a intentar en este Sprint, más el plan para lograrlo.
- **Incremento:** el trabajo terminado que cumple la Definition of Done.

La traducción a Odoo 20

Pieza de Scrum	Dónde vive en Odoo 20	App
Producto	Un proyecto	Proyecto
Product Owner, Scrum Master, Developers	Roles de proyecto (nuevo en Odoo 20)	Proyecto
Historia de usuario	Tarea	Proyecto
Épica	Tarea padre con subtareas, o etiqueta	Proyecto
Sprint	Hito (milestone) con fecha límite	Proyecto
Estados del trabajo	Etapas del Kanban	Proyecto
Story points	Campo personalizado (Studio o módulo)	Studio
Capacidad del equipo	Turnos enlazados a tareas	Planning
Esfuerzo real	Partes de horas y temporizador	Hojas de horas
Daily Scrum	Canal del proyecto y llamadas	Conversaciones
Definition of Done	Artículo de referencia y checklist en la tarea	Conocimiento
Burndown	Informe de burndown del proyecto	Proyecto
Sprint Review	Portal de cliente y valoraciones	Proyecto, Portal
Retrospectiva	Encuestas en el chat y artículo de acta	Conversaciones, Conocimiento

Odoo no tiene un objeto «Sprint» nativo. La convención más limpia es usar **hitos** como Sprints, porque tienen fecha límite, se asignan a tareas y se pueden filtrar y agrupar en todas las vistas. El Apéndice C muestra cómo crear un modelo Sprint propio si el equipo necesita más.

Capítulo 2 — Preparar la cancha: configuración de Odoo 20

Una cancha bien marcada evita discusiones durante el partido. En Odoo eso significa instalar las apps justas, activar cuatro ajustes de Proyecto y dar a cada persona el acceso que necesita, ni más ni menos.

2.1 Apps que hay que instalar

App	Para qué sirve en Scrum	Imprescindible
Proyecto	Backlog, Sprints, tablero Kanban, burndown	Sí
Hojas de horas	Esfuerzo real y temporizador	Sí
Conversaciones (Discuss)	Daily, canal del equipo, encuestas	Sí
Conocimiento	Definition of Done, actas, guías	Recomendada
Planning	Capacidad y turnos ligados a tareas	Recomendada
Studio	Story points y campos propios sin programar	Recomendada (Enterprise)
Hoja de cálculo / Tableros	Velocidad y métricas del equipo	Opcional
CRM y Ventas	Crear el proyecto desde la oportunidad y facturar	Opcional

2.2 Ajustes de Proyecto

En **Proyecto › Configuración › Ajustes** activa:

1. **Hitos (Milestones):** serán los Sprints.
2. **Dependencias de tareas:** para marcar qué historia bloquea a otra.
3. **Valoraciones del cliente:** útiles en la Sprint Review.
4. **Partes de horas** en el proyecto, para comparar lo estimado con lo real.

2.3 Etapas del tablero

Las etapas son las zonas de la cancha. Mantenlas pocas y con significado claro:

Etapa	Qué significa	¿Plegada?
Backlog	Historias candidatas, ordenadas por el Product Owner	No
Listo para Sprint	Refinadas, estimadas y con criterios de aceptación	No
En curso	Alguien está trabajando en ella ahora	No
En revisión	Revisión de código, pruebas o validación funcional	No
Hecho	Cumple la Definition of Done	Sí
Descartado	Ya no aporta valor	Sí

Marca **Hecho** y **Descartado** como etapas plegadas y cerradas, para que el burndown y los filtros de «abiertas» las excluyan.

2.4 Roles de proyecto (novedad de Odoo 20)

Odoo 20 añade un menú de **Roles de proyecto** y permite asignar miembros del equipo a cada rol. Crea tres roles: *Product Owner*, *Scrum Master* y *Developer*. Así el reparto de responsabilidades queda visible en el propio proyecto, y las actividades pueden asignarse a un rol en lugar de a una persona concreta, otra novedad general de esta versión.

2.5 Accesos y usuarios

- **Equipo interno:** usuario interno con permiso de Usuario en Proyecto; el Product Owner, con permiso de Administrador.
- **Usuario ligero (Light User):** Odoo 20 introduce este tipo de acceso limitado al backend. Encaja con perfiles que solo registran horas o actualizan tareas.
- **Cliente o interesado externo:** acceso de portal. En Odoo 20 se le puede dar acceso al proyecto como colaborador sin hacerle seguidor, y se le pueden asignar tareas si sigue el proyecto.

Consejo de entrenador. En Odoo 20 los permisos se definen con el nuevo modelo de accesos con dominio, que sustituye a las reglas de registro clásicas. Si migras un proyecto con reglas personalizadas desde Odoo 19, revísalas antes del primer Sprint.

Capítulo 3 — El draft: construir el Product Backlog

El backlog es la lista de fichajes: ordenada, con informe de cada jugador y revisada antes de cada temporada. En Odoo es simplemente el conjunto de tareas abiertas del proyecto que todavía no tienen Sprint (hito) asignado.

3.1 Anatomía de una historia en Odoo

Campo de la tarea	Uso en Scrum
Título	La historia en formato breve: <i>Como [rol] quiero [acción] para [beneficio]</i>
Descripción	Contexto y criterios de aceptación como lista de comprobación
Prioridad (estrella)	Marca lo urgente; el orden fino se hace arrastrando en la lista
Etiquetas	Épica, tipo (historia, bug, deuda técnica, spike) y módulo afectado
Horas asignadas	Estimación en horas si el equipo no usa puntos
Story points	Campo personalizado (ver 3.3)
Hito	Vacío mientras esté en el backlog; el Sprint cuando entra
Subtareas	El desglose técnico que hacen los Developers
Dependencias	Bloqueos entre historias

3.2 Ordenar el backlog

El Product Owner ordena el backlog en la **vista lista**, agrupada por etapa y filtrada por «sin hito». Odoo 20 permite arrastrar varios registros a la vez en listas y Kanban, lo que acelera mucho el refinamiento: se seleccionan cinco historias y se suben juntas.

Guarda esa búsqueda como favorito. En Odoo 20 los favoritos de lista recuerdan las columnas visibles, y el enlace compartible conserva filtros y agrupación, así que todo el equipo ve exactamente el mismo backlog.

3.3 Añadir story points

Odoo no trae puntos de historia de serie. Hay dos caminos:

1. **Con Studio (sin código):** abre una tarea, entra en Studio y añade un campo de tipo *Selección* llamado «Story points» con los valores de Fibonacci: 1, 2, 3, 5, 8, 13, 21. Muéstralo en la tarjeta Kanban y en la lista.
2. **Con un módulo propio:** un campo entero `x_story_points` en `project.task`. Es la opción buena si quieres calcular la velocidad automáticamente. El Apéndice C incluye el código.

3.4 Refinamiento continuo

Dedica alrededor del 10 % del Sprint a refinar. Una historia está **lista para Sprint** cuando cumple la *Definition of Ready*:

- ☐ Tiene criterios de aceptación comprobables.
- ☐ Está estimada por el equipo.
- ☐ No depende de nada bloqueado.
- ☐ Cabe en un solo Sprint (si no, se parte en subtareas o historias más pequeñas).

Cuando las cumple, se arrastra a la etapa **Listo para Sprint**. Ese movimiento es la señal de que el jugador está en forma para salir a pista.

Capítulo 4 — Salto inicial: Sprint Planning

La Sprint Planning responde a dos preguntas: qué vamos a entregar y cómo lo vamos a hacer. En Odoo se resuelve en una sola pantalla: el hito del Sprint, el tablero filtrado y la capacidad del equipo en Planning.



4.1 Crear el Sprint como hito

1. Abre el proyecto y entra en **Hitos** (desde el botón inteligente o el menú del proyecto).
2. Crea un hito con nombre normalizado: **Sprint 07 – Facturación recurrente**. El número facilita ordenar; el subtítulo recoge el **objetivo del Sprint**.
3. Fija la **fecha límite** en el último día del Sprint, el de la Review.
4. Si el proyecto factura por hitos, vincula el hito a la línea de pedido correspondiente (ver Capítulo 9).

4.2 Medir la capacidad

Antes de elegir historias, el equipo calcula cuánto puede abarcar. Dos fuentes en Odoo 20:

- **Velocidad histórica:** puntos terminados en los tres últimos Sprints (Capítulo 6).
- **Disponibilidad real:** la app Planning muestra turnos, ausencias y festivos. En Odoo 20 un turno puede enlazarse a una tarea concreta, y desde la tarea el botón *Planificar* la lleva a Planning. También se pueden asignar varios recursos a un turno y dar prioridad a los turnos para que la planificación automática coloque primero lo importante.

Una fórmula sencilla para empezar:

$$\text{Capacidad}_{\{\text{Sprint}\}} = \text{Velocidad}_{\{\text{media}\}} \times \frac{\{\text{Días disponibles}\}}{\{\text{Días del Sprint con equipo completo}\}}$$

4.3 Seleccionar historias

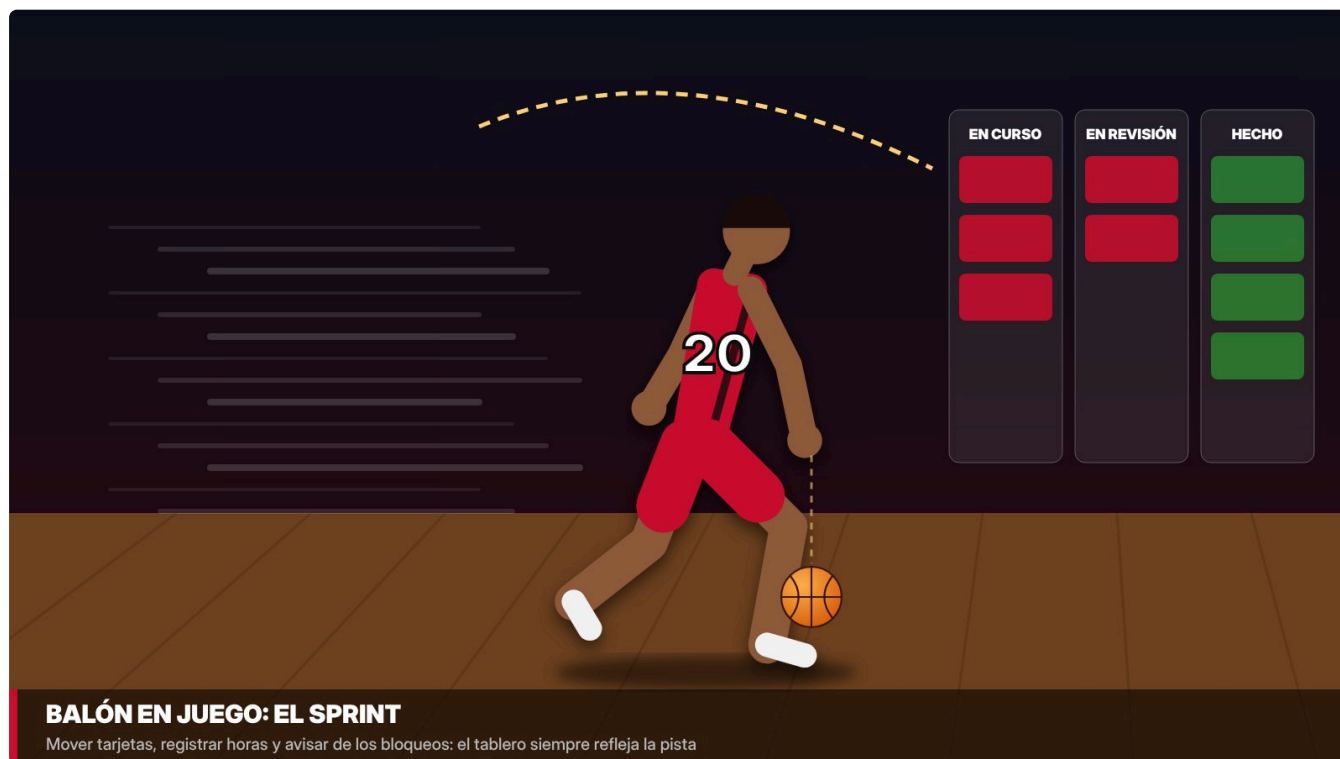
1. Filtra el tablero por etapa **Listo para Sprint** y agrupa por prioridad.
2. Selecciona las historias que caben en la capacidad y asígnales el hito del Sprint de una vez con la edición múltiple de la lista.
3. Los Developers parten cada historia en **subtareas técnicas** y se las asignan.
4. Revisa el **Gantt** del proyecto: en Odoo 20 se puede seleccionar una fila o columna entera, arrastrar tareas para reprogramar e imprimir el calendario de tareas.

4.4 Cerrar la Planning

Al salir de la reunión deben existir en Odoo tres cosas: el hito con su objetivo y fecha, todas las historias del Sprint asignadas a ese hito y con responsable, y un mensaje fijado en el chatter del proyecto con el objetivo del Sprint. Odoo 20 permite fijar mensajes importantes en el chatter, y es el sitio ideal para que nadie lo pierda de vista.

Capítulo 5 — Balón en juego: ejecutar el Sprint

Durante el Sprint, Odoo debe reflejar la realidad en todo momento: si una tarjeta no se mueve, el trabajo no se está haciendo o no se está contando. La disciplina es simple: mover tarjetas, registrar horas y hablar en el canal.



5.1 El tablero del Sprint

Crea un favorito «Sprint actual»: vista Kanban, filtro por el hito vigente, agrupado por etapa. Reglas del juego:

- **Una persona, una tarea en curso.** Si alguien tiene tres tarjetas en *En curso*, hay un problema de foco.
- **Limita el trabajo en curso.** Odoo no impone límites WIP por etapa, pero puedes mostrar el contador de la columna y acordar un máximo en la Definition of Done del equipo.
- **Bloqueado = visible.** Usa el estado de la tarea (bloqueado, en rojo) y deja en el chatter qué lo bloquea. La dependencia entre tareas hace el resto.

5.2 El Daily Scrum en Odoo

Crea un canal en Conversaciones con el nombre del proyecto. Para el Daily:

1. Videollamada desde el propio canal a la misma hora cada día, 15 minutos como máximo.
2. El Scrum Master comparte pantalla con el tablero del Sprint, no con una presentación.
3. Cada Developer mueve sus tarjetas en directo mientras habla.

4. Los bloqueos se apuntan como actividad asignada al Scrum Master.

En Odoo 20 las llamadas pueden **grabarse con un resumen** de lo hablado, y los canales se organizan en categorías y favoritos. Útil para equipos distribuidos o para quien no pudo conectarse ese día.

5.3 Registrar el esfuerzo

El esfuerzo real alimenta la estimación del siguiente Sprint, así que conviene que registrarlo cueste poco. Odoo 20 lo pone fácil:

- **Temporizador en la barra superior:** se inicia desde cualquier pantalla de Odoo, se elige proyecto y tarea, y se marca si es facturable.
- **Asistente de hojas de horas:** sugiere entradas a partir de la actividad del usuario. Las reglas pueden compartirse con el equipo y existe la opción de *actividad secundaria* para interrupciones rápidas.
- **Integración con ActivityWatch:** convierte la actividad del dispositivo en partes de horas manteniendo los datos en privado.
- **Edición incremental de duraciones:** en cualquier campo de duración se puede escribir `+= 0.5` para sumar media hora sin calcular a mano.

5.4 Trabajar sin conexión y desde el móvil

Odoo 20 trae un modo sin conexión que permite crear y editar registros sin red, y una interfaz móvil rediseñada. Un Developer en una obra o en casa de un cliente puede mover su tarea y registrar horas aunque la cobertura falle; los cambios se sincronizan al volver.

5.5 Proteger el Sprint

Una vez empezado, el Sprint no se toca desde fuera. Si llega una petición urgente, entra como tarea **sin hito** en el backlog y el Product Owner decide. Solo si es una emergencia real se negocia sacar otra historia de igual tamaño, y queda anotado en el chatter del hito.

Capítulo 6 — El marcador: métricas del equipo

Un equipo Scrum necesita tres números, no veinte: lo que queda por hacer en el Sprint (burndown), lo que suele terminar por Sprint (velocidad) y cuánto se desvía lo real de lo estimado. Odoo 20 da los tres sin salir del sistema.

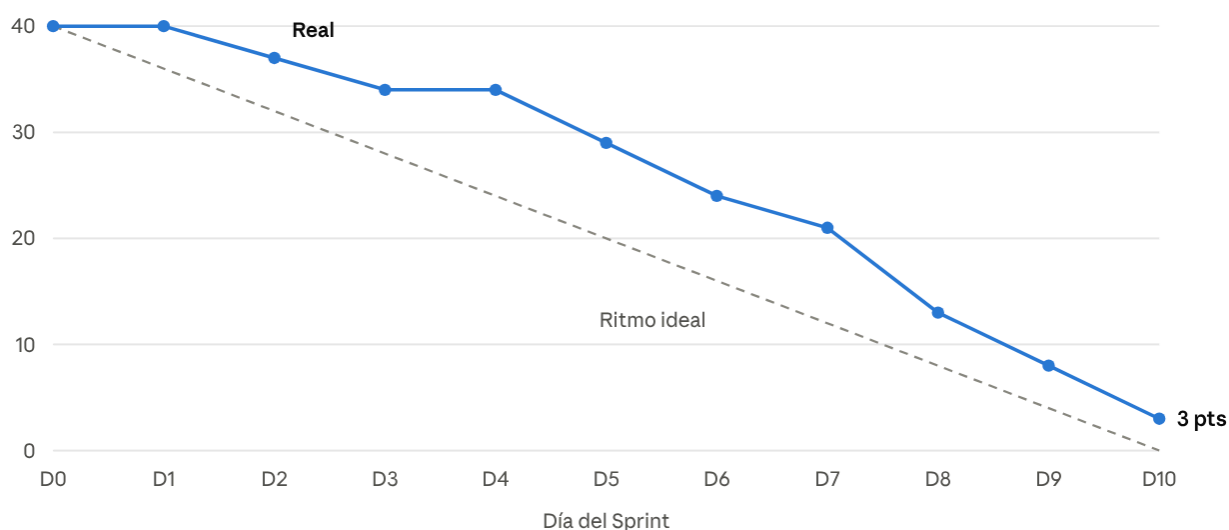


6.1 Burndown

El proyecto incluye un informe de **burndown** en sus análisis. Filtra por el hito del Sprint y mide por número de tareas o, si añadiste el campo, por story points. En Odoo 20 el seguimiento de cambios pasó a generarse en el propio mensaje, y Odoo adaptó el burndown y la duración por etapa a ese cambio; si una auditoría necesita los valores de seguimiento antiguos, existe un módulo específico para conservarlos.

El Sprint de ejemplo termina con 3 puntos sin cerrar

Story points pendientes por día de un Sprint de 10 días (ejemplo ilustrativo)



Ejemplo ilustrativo · Sprint de 10 días y 40 story points

Cómo leerlo: si la línea real va por encima de la ideal durante la primera mitad, el equipo arrancó tarde o entraron historias demasiado grandes. Una meseta de varios días suele ser un bloqueo que no se ha hecho visible en el tablero.

6.2 Velocidad

La velocidad es la suma de story points en etapa **Hecho** por hito. En la vista pivote de tareas: filas = hito, medida = story points, filtro = etapa plegada «Hecho». Para seguirla en el tiempo, inserta ese pivote en una **hoja de cálculo** de Odoo; en Odoo 20 los pivotes admiten dominio y estilos, y hay nuevos tipos de gráfico y anotaciones. Usa la media de los tres últimos Sprints para planificar, nunca el mejor.

6.3 Estimado frente a real

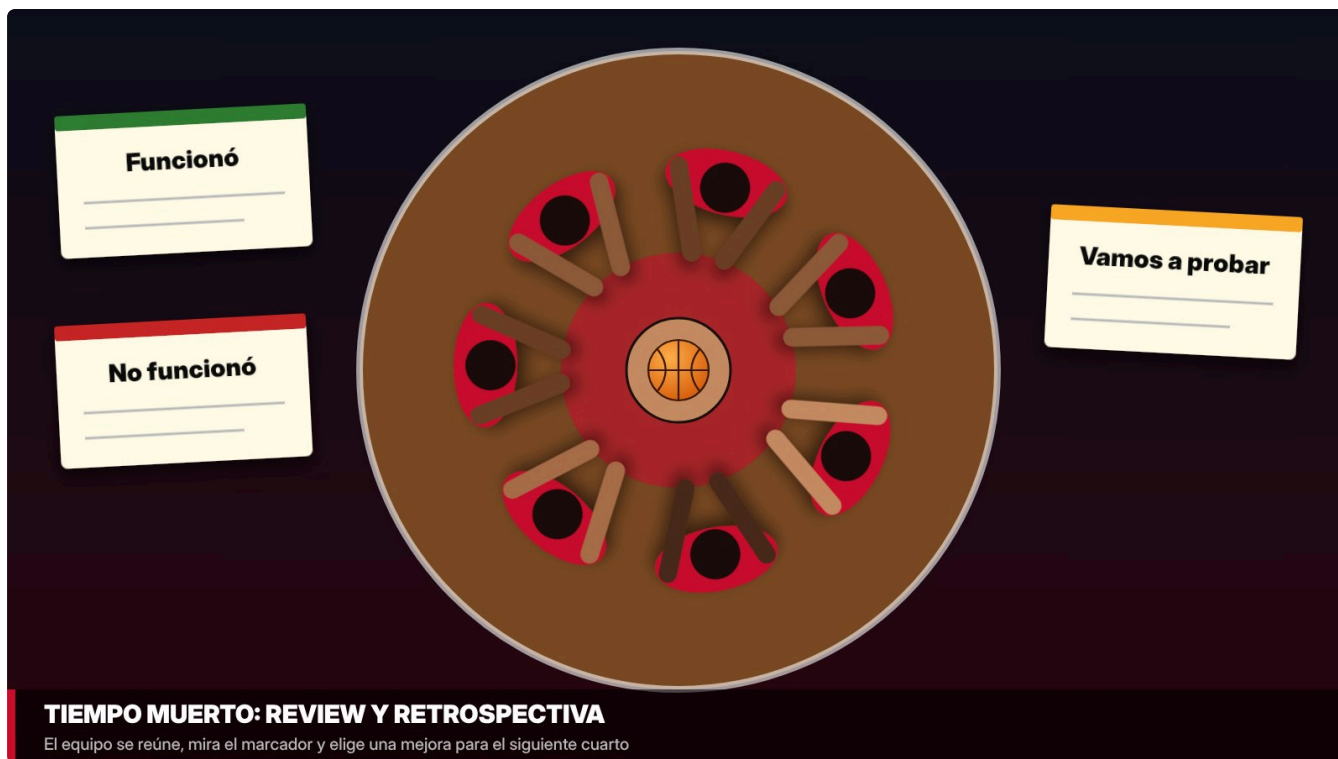
Compara **horas asignadas** con **horas registradas** por tarea y por Developer en el análisis de hojas de horas. El tablero de Hojas de horas de Odoo 20 también compara objetivos de facturación con tiempo facturable, útil si el proyecto se cobra por horas.

6.4 Un tablero para el equipo

Crea un **tablero (Dashboard)** con el burndown del Sprint actual, la velocidad de los últimos Sprints y las tareas bloqueadas. En Odoo 20 los tableros pueden ser privados y compartirse con usuarios concretos, y los filtros globales pueden guardarse como favoritos con un filtro por defecto. Compártelo con el Product Owner y con el cliente si procede.

Capítulo 7 — Tiempo muerto: Review y Retrospectiva

La Review mira el producto; la Retrospectiva mira al equipo. Son dos tiempos muertos distintos y en Odoo conviene que dejen rastros distintos: la Review en el proyecto y el portal, la Retrospectiva en Conocimiento.



7.1 Sprint Review con el cliente

1. **Antes:** filtra las tareas del hito en etapa *Hecho*. Esa lista es el orden del día.
2. **Durante:** enseña el incremento funcionando en una base de datos de pruebas, no capturas de pantalla.
3. **Portal:** en Odoo 20 el cliente puede acceder al proyecto como colaborador y ve en el portal los cambios del chatter de cada tarea. Puede comentar directamente sobre la historia entregada.
4. **Valoraciones:** si activaste las valoraciones del cliente, pídele que valore las tareas cerradas al pasar a *Hecho*.
5. **Después:** lo que el cliente pida nuevo entra al backlog como tarea sin hito. El Product Owner la ordena en el siguiente refinamiento.
6. **Cierre del hito:** marca el hito como alcanzado. Si factura por hitos, ese es el disparador de la factura (Capítulo 9).

7.2 Sprint Retrospective en el vestuario

Usa una plantilla en **Conocimiento** con tres bloques: *qué funcionó*, *qué no funcionó* y *qué vamos a probar*. Duplica el artículo para cada Sprint y enlázalo desde el chatter del hito.

Para votar mejoras, Odoo 20 permite crear **encuestas dentro de un canal de Conversaciones** con varias opciones y duración limitada. El equipo vota en dos minutos la acción de mejora prioritaria.

7.3 Que la mejora no se quede en el acta

Cada acción de mejora se convierte en una **tarea del Sprint siguiente**, con responsable y etiqueta *Mejora*. Si no tiene tarea, no existe. En la próxima retrospectiva lo primero es revisar si esas tareas llegaron a *Hecho*.

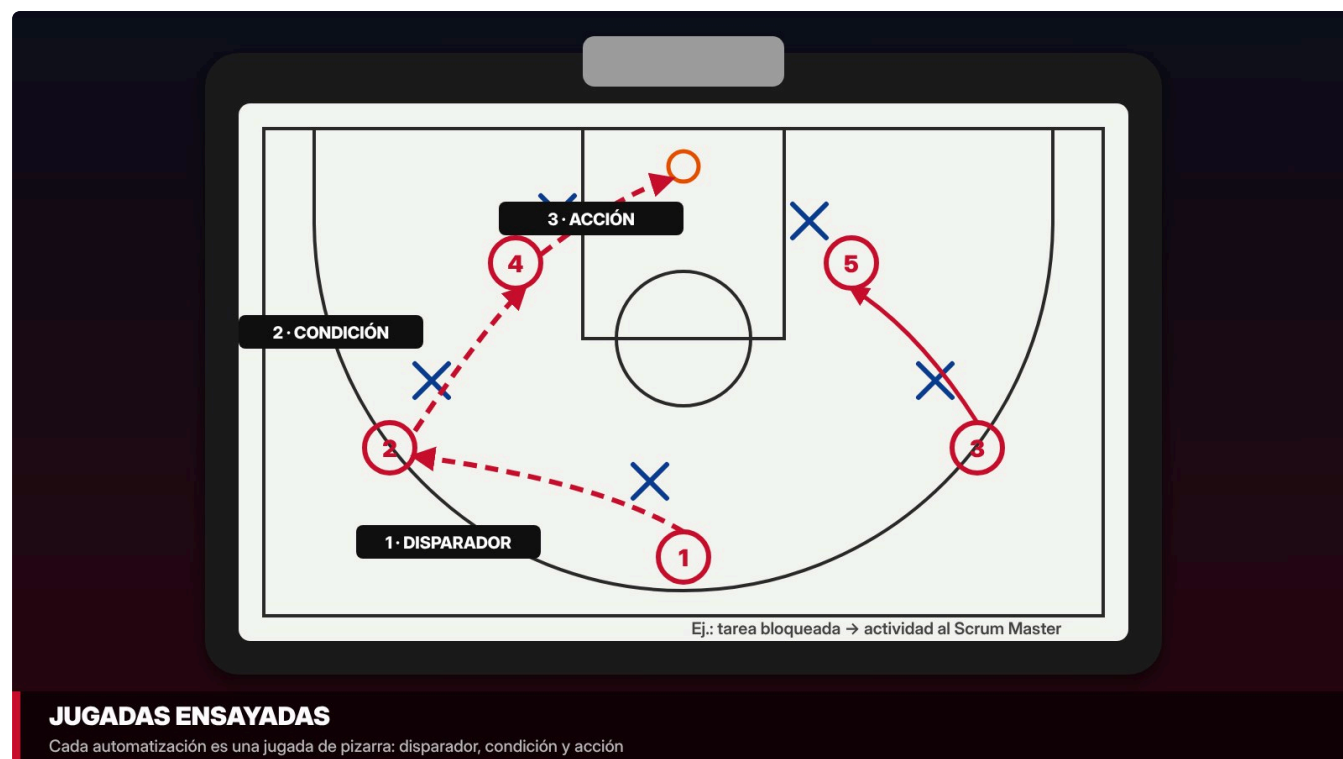
7.4 Formatos de retrospectiva por temporada

Formato	Cuándo usarlo	Columnas en el artículo
Bien / Mal / Probar	Equipos nuevos, por defecto	3
Start / Stop / Continue	Cuando hay hábitos claros que cambiar	3
4L (Liked, Learned, Lacked, Longed for)	Tras un Sprint largo o complejo	4
Barco de vela (viento, anclas, rocas, isla)	Para hablar de riesgos y objetivos	4

Cambiar de formato cada pocos Sprints evita que la retrospectiva se vuelva rutina.

Capítulo 8 — Jugadas ensayadas: automatizaciones, Studio e IA

Las jugadas ensayadas ahorran decisiones repetidas. En Odoo 20 hay tres niveles: reglas de automatización sin código, Studio para adaptar pantallas y los nuevos agentes de IA para el trabajo que antes requería a una persona.



8.1 Automatizaciones que todo equipo Scrum debería tener

Disparador	Acción	Por qué
Tarea pasa a <i>En curso</i>	Asignar al usuario actual si no tiene responsable	Nadie trabaja en tareas «de nadie»
Tarea pasa a <i>Hecho</i> sin horas registradas	Crear actividad para el responsable	Los datos de esfuerzo no se pierden
Tarea marcada como bloqueada	Actividad para el rol <i>Scrum Master</i>	El bloqueo llega a quien debe quitarlo
Fecha límite del hito mañana	Aviso en el canal del proyecto	Preparar la Review a tiempo
Tarea con etiqueta <i>Bug</i> creada desde el portal	Etapas <i>Backlog</i> y prioridad alta	El Product Owner la ve en el refinamiento

Se configuran en **Ajustes › Técnico › Reglas de automatización** (modo desarrollador) o desde Studio. En Odoo 20 una regla también puede lanzar **planes de actividades** y actividades asignadas a un rol.

8.2 Studio para el tablero Scrum

- Añade *Story points* y *Valor de negocio* a la tarjeta Kanban.

- Reordena pestañas y botones del formulario arrastrando, una novedad de Studio en Odoo 20.
- Usa la propiedad *Aviso* en campos de fecha para marcar en rojo las tareas con fecha límite vencida.
- Añade un campo *Criterios de aceptación* de tipo HTML si prefieres separarlos de la descripción.

8.3 Agentes de IA en el Sprint

Odoo 20 convierte la IA en agentes que crean y actualizan registros, incluso a partir de un archivo de instrucciones como un PDF. Pueden lanzarse desde acciones automatizadas o programadas, se les habla por voz y piden confirmación mediante botones. Las funciones de IA consumen créditos IAP. Usos prácticos en Scrum:

1. **Del documento al backlog:** subes el pliego o el acta de requisitos del cliente y el agente propone historias de usuario como tareas en etapa *Backlog*. El Product Owner las revisa antes de ordenar.
2. **Resumen del Sprint:** un agente programado el último día resume las tareas cerradas del hito para la Review.
3. **Primeros criterios de aceptación:** a partir del título de la historia, el agente sugiere criterios que el equipo corrige.
4. **Conexión externa:** Odoo 20 se conecta a herramientas externas por MCP (Model Context Protocol), lo que permite enlazar el backlog con el repositorio de código o con otros asistentes.

Regla de oro. La IA propone; el equipo decide. Ninguna historia generada por un agente entra en un Sprint sin pasar por el refinamiento.

Capítulo 9 — Liga profesional: escalar Scrum y conectar con el negocio

Cuando hay varios equipos o el proyecto se vende y se factura, Scrum tiene que hablar con Ventas y Contabilidad. Odoo 20 cierra ese círculo: de la oportunidad al proyecto, del hito a la factura.

9.1 De la oportunidad al proyecto

En Odoo 20 se puede **crear un proyecto directamente desde una oportunidad del CRM** eligiendo una plantilla de proyecto. Prepara una plantilla «Proyecto Scrum» con:

- Las seis etapas del Capítulo 2.

- Los tres roles de proyecto definidos.
- Hitos vacíos para los primeros cuatro Sprints.
- Una tarea fija *Sprint 0: preparación del entorno* con su checklist.

Así cada proyecto ganado arranca con el mismo reglamento sin configurarlo a mano.

9.2 Facturar con Scrum

Modelo de contrato	Configuración en Odoo 20	Encaje con Scrum
Precio por Sprint	Producto de servicio facturado por hitos; un hito por Sprint	El más natural: se cobra lo entregado y aceptado en cada Review
Tiempo y materiales	Servicio facturado según partes de horas	Flexible, pero exige disciplina con el temporizador
Bolsa de horas prepagada	Prepago y cargo de excesos desde el asistente de facturación	Bueno para soporte evolutivo tras el go-live
Precio cerrado	Servicio de cantidad fija con entregas parciales	Arriesgado: el alcance fijo choca con el backlog vivo

Para seguir la rentabilidad, Odoo 20 sustituye el antiguo informe de rentabilidad del tablero del proyecto por informes dedicados de **presupuesto** y de **márgenes reales y previstos**.

9.3 Varios equipos, un producto

- **Un proyecto por equipo** y una etiqueta común por producto, o **un proyecto por producto** con un rol *Developer* por equipo y filtros por responsable. La primera opción escala mejor a partir de tres equipos.
- **Scrum de Scrums**: un canal compartido en Conversaciones y un tablero que agrupa tareas bloqueadas de todos los proyectos.
- **Dependencias entre equipos**: se modelan como dependencias de tareas entre proyectos, visibles en el Gantt.
- **Sprints sincronizados**: todos los equipos usan las mismas fechas de hito para que la Review sea conjunta.

9.4 Después del go-live: Scrum + Helpdesk

Una vez en producción, las incidencias entran por **Helpdesk**. En Odoo 20 la IA detecta tickets similares para sugerir respuestas, y hay un filtro para los tickets en los que el último mensaje es del cliente. Los tickets que requieren desarrollo se convierten en tareas del backlog con la etiqueta *Bug*, y el Product Owner decide si entran en el próximo Sprint.

Capítulo 10 — Partido real: un caso práctico completo

Este capítulo sigue un proyecto ficticio de principio a fin para ver todas las piezas juntas. *Distribuciones Turia* es una distribuidora que implanta Odoo 20 (Ventas, Inventario y Contabilidad) con un consultor y dos desarrolladores, en Sprints de dos semanas.

10.1 Plantilla de partida

- **Oportunidad ganada en CRM** → se crea el proyecto con la plantilla «Proyecto Scrum».
- **Roles:** el consultor es Scrum Master; el director de operaciones del cliente es Product Owner con acceso de portal como colaborador; los dos desarrolladores son Developers.
- **Sprint 0 (una semana):** base de datos de pruebas, importación de contactos y productos, Definition of Done redactada en Conocimiento.

10.2 El backlog inicial

El Product Owner sube el documento de requisitos y un agente de IA propone 34 historias. En el refinamiento el equipo descarta 6, parte 4 en historias más pequeñas y estima el resto con story points.

10.3 Plan de Sprints

Sprint (hito)	Objetivo	Historias clave
Sprint 01 – Vender	Un pedido de venta real de principio a fin	Tarifas, plantillas de presupuesto, confirmación por email
Sprint 02 – Entregar	Del pedido al albarán y la entrega	Rutas, reglas de reabastecimiento, etiquetas
Sprint 03 – Cobrar	Factura, cobro y conciliación bancaria	Diarios, reglas de conciliación, recordatorios de pago
Sprint 04 – Arrancar	Migración final, formación y go-live	Saldo iniciales, usuarios, checklist de arranque

10.4 Lo que pasó en el Sprint 02

- **Día 3:** el burndown se queda plano. En el Daily se ve que la historia de rutas depende de una decisión del cliente. Se marca como bloqueada, la automatización crea una actividad al Scrum Master y la decisión llega al día siguiente por el portal.
- **Día 6:** llega una petición urgente de etiquetas personalizadas. Entra como tarea sin hito; el Product Owner la deja para el Sprint 03.
- **Review:** el cliente valida 5 de 6 historias desde el portal. La sexta vuelve al backlog con un comentario concreto.
- **Retrospectiva:** la encuesta del canal elige «revisar dependencias con el cliente en la Planning». Se crea como tarea con etiqueta *Mejora* en el Sprint 03.
- **Facturación:** el hito del Sprint 02 se marca como alcanzado y se emite la factura del Sprint.

10.5 Lecciones del partido

1. El bloqueo más caro fue una decisión de negocio, no técnica: invitar al Product Owner al Daily una vez por semana lo resolvió.
2. La facturación por hitos alinea a todos: el cliente acepta en la Review y el equipo cobra lo que entrega.
3. El temporizador de Odoo 20 hizo que el equipo registrara horas el mismo día, y la estimación del Sprint 03 fue más precisa.

Apéndices

Apéndice A — Checklist de arranque

- ☐ Apps instaladas: Proyecto, Hojas de horas, Conversaciones, Conocimiento, Planning.
- ☐ Hitos, dependencias y valoraciones activados en los ajustes de Proyecto.
- ☐ Seis etapas creadas; *Hecho* y *Descartado* plegadas.
- ☐ Roles de proyecto: Product Owner, Scrum Master, Developer.
- ☐ Campo *Story points* añadido (Studio o módulo).
- ☐ Favoritos guardados: *Backlog* y *Sprint actual*.
- ☐ Canal del proyecto creado en Conversaciones.
- ☐ Definition of Ready y Definition of Done publicadas en Conocimiento.
- ☐ Primer hito creado con objetivo y fecha límite.
- ☐ Reglas de automatización del Capítulo 8 activas.
- ☐ Tablero de métricas compartido con el Product Owner.

Apéndice B — Glosario Scrum ↔ Odoo

Término Scrum	En Odoo 20
Sprint	Hito del proyecto
Product Backlog	Tareas abiertas sin hito
Sprint Backlog	Tareas del hito vigente
Historia de usuario	Tarea
Épica	Tarea padre o etiqueta
Incremento	Tareas del hito en etapa <i>Hecho</i>
Impedimento	Tarea en estado bloqueado + actividad
Velocidad	Pivote de story points por hito
Definition of Done	Artículo de Conocimiento + checklist

Apéndice C — Módulo de ejemplo: story points y Sprints

Un punto de partida mínimo que añade story points a las tareas y totales al hito. Solo hereda modelos existentes, así que no necesita reglas de acceso nuevas. Pruébalo en una base de datos de desarrollo de Odoo 20 antes de usarlo, porque los nombres de vistas pueden cambiar entre versiones.

`__manifest__.py`

```
{
    "name": "Scrum Story Points",
    "version": "20.0.1.0.0",
    "summary": "Story points en tareas y totales por Sprint (hito)",
    "depends": ["project"],
    "data": ["views/project_views.xml"],
    "license": "LGPL-3",
}
```

`models/project_scrum.py`

```
from odoo import api, fields, models

FIBONACCI = [1, 2, 3, 5, 8, 13, 21]

class ProjectTask(models.Model):
    _inherit = "project.task"

    story_points = fields.Integer("Story points", aggregator="sum")

    @api.constrains("story_points")
    def _check_story_points(self):
        for task in self:
            if task.story_points and task.story_points not in FIBONACCI:
                raise models.ValidationError(
                    "Usa un valor de Fibonacci: 1, 2, 3, 5, 8, 13 o 21."
                )

class ProjectMilestone(models.Model):
    _inherit = "project.milestone"

    sprint_goal = fields.Char("Objetivo del Sprint")
    committed_points = fields.Integer(
        "Puntos comprometidos", compute="_compute_points", store=True
    )
    done_points = fields.Integer(
        "Puntos terminados", compute="_compute_points", store=True
    )

    @api.depends("task_ids.story_points", "task_ids.stage_id.fold")
    def _compute_points(self):
        for sprint in self:
```

```

        tasks = sprint.task_ids
        sprint.committed_points = sum(tasks.mapped("story_points"))
        sprint.done_points = sum(
            tasks.filtered(lambda t:
                t.stage_id.fold).mapped("story_points")
        )

```

views/project_views.xml

```

<odoo>
    <record id="view_task_form_story_points" model="ir.ui.view">
        <field name="name">project.task.form.story.points</field>
        <field name="model">project.task</field>
        <field name="inherit_id" ref="project.view_task_form2"/>
        <field name="arch" type="xml">
            <field name="milestone_id" position="after">
                <field name="story_points"/>
            </field>
        </field>
    </record>
</odoo>

```

Con este módulo, la velocidad sale directamente de una vista lista o pivote de hitos con las columnas *Puntos comprometidos* y *Puntos terminados*. La regla «tarea terminada = etapa plegada» sigue la convención del Capítulo 2; si tu equipo usa el estado de la tarea para marcar «hecho», cambia el filtro.

Fuentes

- [Novedades de Odoo 20, app por app \(Cybrosys\)](#)
- [Odoo 20: actualizaciones clave explicadas \(Cybrosys\)](#)

EL PARTIDO SE GANA EN EQUIPO

Una guía práctica para aplicar Scrum con las herramientas que ya tienes en Odoo 20: proyecto, hitos, hojas de horas, Planning, Conocimiento y agentes de IA.

EN ESTE LIBRO

- Backlog vivo y Sprints con hitos
- Daily, Review y Retro en Odoo
- Burndown, velocidad y tableros
- Automatizaciones e IA
- Facturar por Sprint
- Caso real de implantación

