

Requisitos Sprint 2 – Practica I

El siguiente documento tiene como objetivo que los equipos apliquen los 5 eventos principales de Scrum (sprint, Planning, Daily, Review, Retrospective, y Refinement) de forma adaptada a su contexto: desarrollo de flujos AI en n8n, trabajo en parejas (developer–tester), y evidencias verificables.

Integra guías y formatos para estudiantes de Práctica I UNAB; que trabajan con flujos de AI en n8n. Este documento provee plantillas listas para usar en Taiga: Retrospectiva, Refinamiento, Pruebas Unitarias e Integración, y recomendaciones de evidencia.

Recordar que los sprint duran 3 semanas y que es el equipo el que organiza qué hacer semanalmente. Inicio del sprint 1: **13 de octubre al 02 de noviembre, entrega de video el 22 de noviembre. Revisión de incremento 29 de noviembre**

Sprint 2 Inicio: 03 de noviembre al 23 de noviembre

Sprint 3 Inicio: 24 de noviembre al 13 de diciembre. Revisiones por definir

Uso de Taiga: recomendaciones

1. Crear una Wiki del Sprint (Sprint 1) y subir allí los documentos PDF finalizados.
2. En cada HU debe desarrollarse detalladamente usando **Definition of Ready (DoR)**
3. Por cada tarea definida para una Historia adjuntar evidencia: capturas de n8n, JSON exportado del flujo, registros de pruebas, imágenes, archivos, datos, etc)
4. Vincular la HU con la Tarea correspondiente y marcar estados con claridad (Nueva, En curso, En revisión, Hecha).
5. Subir los informes solicitados. Recuerde que el compromiso es trabajar 10 horas semanales. Que muchos grupos no han cumplido. Debe cumplir el total de horas de la práctica.
6. Recordar que el proyecto genera evidencias por tarea, cada evidencia por tarea también es un informe
7. Se recomienda que se usen Historias Técnicas para declarar las tareas de aspectos técnicos como Usar git, que incluya las tareas como instalación y uso de git
8. También se puede usar Historias de Proyecto, donde se incluyan las tareas derivadas de los eventos del scrum (planning, dailys, review, refinamiento, retrospectiva)
9. Recuerde que debe lograr evidenciar 216 horas en el desarrollo de todo el proyecto, desde que recibió el proyecto, asistió a la primera reunión y comenzó a capacitarse en las tecnologías asociadas al proyecto
10. Desde este sprint, todas las tareas de taiga pueden verse reflejadas en su archivo Excel en donde se hace el seguimiento de las 216 horas de trabajo por alumno.

Requisitos Sprint 2 – Practica I

1. Evento Sprint Planning

Objetivos del evento:

Definir el **objetivo del Sprint 2**. Entregar un flujo funcional integrado de todas las historias que ingresarán a este sprint por tener la mayor priorización. Debe ingresar al menos 3 HU.

- Asignar a menos una HU por developer y vincular un tester (PO o SM).
- Elaborar el **Sprint Backlog** con tareas concretas por HU y responsables. Debe usar Taiga para representar el sprint backlog
- Aplicar **Planning Poker** para estimar el esfuerzo de la HU. Estudiar el concepto de HU Pivote para mejorar la estimación del equipo
- Definir el documento **DoD**, que incluya al menos lo siguiente:
 1. Objetivo del sprint 2
 2. Integración de los criterios de aceptación de todas las historias
 3. Casos de prueba de los criterios de aceptación
 4. Calidad técnica mínima del flujo: errores controlados, logs, seguridad de credenciales, rendimiento, datos anonimizados, nomenclatura de nodos (AUTH_, API_, MAP_, CTRL_, OUT_).

Recomendaciones:

- Cada HU debe estar **completa y ejecutable**, aunque repita parte del flujo de otro integrante.
- Documentar las tareas derivadas en formato tabla:
 - ID HU, descripción, tareas, responsable, estimación, estado inicial, todo esto en taiga
- Utilizar una herramienta de estimación:
 - ⇒ Planning Poker Online – [Scrumpoker-online.org](https://www.scrumpoker-online.org)
 - (recomienda usar valores Fibonacci 1–2–3–5–8–13 y explicar por qué eligieron la estimación final).
- Cada HU debe crear, ejecutar y documentar al menos 3 casos de prueba por cada criterio de aceptación (CA)
- Al terminar de desarrollar y probar las HU, es esperable que tengan código o funcionalidades repetidas, por lo tanto en la fase de integración de las HU al flujo final del sprint o incremento, debe quedar definido en el entorno de trabajo del developer líder. Este flujo debe subirse a git y probarse con casos de pruebas de integración.
- Antes del evento de cierre del sprint o retrospectiva, debe hacerse el refinamiento, después de ello debe subirse el flujo integrado

Requisitos Sprint 2 – Practica I

- En este sprint es OBLIGATORIO subir el incremento1 a git del proyecto y importarse el flujo integrado y probado en todos los entornos de programación de los developers y tester. Puede subir las versiones intermedias que se fueron programando, ej. Se permita una HU y se aplican las pruebas unitarias y luego se sube a git; al terminar las n HU, se integran y se sube la primera versión del incremento.

Se aplican pruebas de integración (con más datos de prueba) y se sube la siguiente versión del incremento probada

Informe post-Planning (entregable):

1. Nombre del equipo, roles y HU asignadas. Definición del Lider de developer
2. Objetivo del sprint.
3. Sprint Backlog con tareas por HU.
4. Captura o registro del Planning Poker.
5. Supuestos o dudas no resueltas para el PO.
6. Documentación de cada flujo n8n por HU asignada
7. Registros de pruebas unitarias por cada CA de cada HU
8. Criterios de integración del flujo o incremento
9. Evidencia de reunión con PO externo (Eduardo Navarro)

2. Evento Daily Scrum

Objetivos del evento:

El equipo de desarrollo inspeccione el progreso hacia el objetivo del Sprint y adapte el plan (Sprint Backlog) para las próximas 24–48 horas, identificando y gestionando impedimentos. No es un reporte al SM/PO; es principalmente para gestionar el proyecto y ayudar al control de los Developers.

En concreto, la Daily sirve para:

1. **Inspeccionar:** ¿Cómo vamos respecto al **Objetivo del Sprint**?
2. **Adaptar:** ¿Qué cambiaremos hoy/mañana en el **plan de trabajo** (tareas, prioridades, pares)?
3. **Desbloquear:** ¿Qué **impedimentos** existen y quién los tomará?
4. **Actualizar:** tablero/taiga, tiempos, estado de HU, y compromisos inmediatos.

Requisitos Sprint 2 – Practica I

Se solicita:

Se realizará cada dos días (no es necesario diario a no ser que el equipo lo considere). Deben ser por el mismo medio (presencial, online.. etc) y en la misma hora

Debe durar máximo 15 minutos y registrar/hablar brevemente:

- Qué hice desde el último Daily.
- Qué haré hasta el próximo.
- Qué impedimentos encontré.

El Scrum Master registra y sube capturas o minutas breves.

Cada Developer actualizan taiga y planificación de tareas del sprint backlog

Al final del Sprint se entregará un resumen de Daily.

Para llevar los distintos dailys en el taiga, recomiendo crear una Historia de Negocio (HN), en donde pueda registrar cada reunión.

3. Evento Sprint Review

Objetivos del evento:

Inspeccionar el incremento entregado (HU integradas) funcionando y recoger feedback del PO y adaptar el Product Backlog (qué sigue / qué cambiar)

1. Reunión con Eduardo Navarro.

En esta iteración deben planificar una reunión con Eduardo Navarro, cuando el incremento 2 se encuentre integrado con el incremento 1 y con las HU del sprint 2. Después de las pruebas de integración, mostrar el incremento a Eduardo Navarro. Las observaciones de esta reunión deben entrar al proceso de refinamiento del incremento 2

2. Informes de a Review 2

Acta con las observaciones de Eduardo Navarro y detalle de los acuerdos y observaciones al incremento

Detallar de la reunión: asistentes, duración, lugar y recursos usados en la reunión. Debe estar firmada por Eduardo Navarro.

Para mostrar el incremento debe asegurarse de mostrar los criterios de aceptación y describir las observaciones de Eduardo, para cada uno de los comportamientos funcionales derivados de los CA.

Documentar las observaciones en un formato como el propuesto a continuación:

Requisitos Sprint 2 – Practica I

HU	CA (Given/When/Then)	Observaciones PO	Resultado
			Aceptado Aceptado con observaciones Rechazado Otro
HU-01	AC1: Explicación		
HU-01	AC2: Explicación		

1. **Necesidades para el próximo sprint, o para continuar con este sprint, para antes de entregar**
 - Dato/Regla a confirmar
 - Alternativa A/B propuesta por el equipo.
 - Plazo deseado de respuesta e impacto si no se decide.
2. **Supuestos del sprint actual**
 - Supuesto → Impacto → Evidencia de por qué se asumió.
3. **Acta de la revisión con observaciones de Eduardo**

Requisitos Sprint 2 – Practica I

5. Evento Retrospectiva (Sprint Retrospective)

Objetivos del evento:

Inspeccionar **cómo trabajó el equipo (personas, procesos, herramientas, comunicación)**,

Definir mejoras concretas (**experimentos/acciones**) para el siguiente sprint.

Responsable: Scrum Master. Duración sugerida de la reunión: 30–40 minutos. Subir el acta a la Wiki del Sprint.

Esta segunda retrospectiva debe evaluar si se cumplieron el o los compromisos asumidos en la retrospectiva del sprint1

Sección	Contenido esperado	Ejemplo / Nota
Datos generales	Equipo, Sprint, Fecha, Asistentes.	Equipo A, Sprint 1, 10/10/2025, 5 asistentes.
Lo que salió bien	3–5 fortalezas del Sprint.	Comunicación por Discord, HU entregadas a tiempo.
Lo que podemos mejorar	3–5 puntos de mejora priorizados.	Estimación conservadora; falta de preguntas al PO.
Acciones de mejora	Acciones SMART con responsable y fecha.	Estandarizar nombres de nodos n8n (Resp: Ana, 14/10).
Riesgos y bloqueos	Riesgos detectados y mitigaciones.	Dependencia de API externa; crear mock local.
Compromisos para Sprint 2	1–3 compromisos concretos.	Hacer Daily cada 2 días con minuta breve.
Cumplimiento de compromisos del sprint anterior		Se mejoró la comunicación del equipo, todos asisten a las daily y hablan de su trabajo diario

Requisitos Sprint 2 – Practica I

5. Evento Refinamiento (Backlog Refinement / Grooming)

Objetivos del evento:

Elevar la calidad de los ítems del backlog para que estén READY al entrar al próximo sprint. Estos objetivos están definidos en el DoD, y deben ser caros, estimados y medibles (Casos de prueba).

Responsable: Product Owner (coordina), SM facilita. Objetivo: preparar HU candidatas para el próximo sprint.

Este evento debe ocurrir después de la review y se debe encargar de refinar en tareas, nuevas HO o mejoras del incremento; derivados de las observaciones descritas en el acta de la review.

Evidencias del refinamiento: Resumen de las tareas del Refinamiento:

Entregable esperado	Descripción breve
Backlog actualizado	Historias revisadas, priorizadas y estimadas
Nuevas historias creadas	Basadas en recomendaciones del review
Historias divididas o corregidas	Para mejor ejecución técnica
Sprint Goal preliminar	Definido tentativamente
Minuta de refinamiento	Síntesis de cambios y acuerdos

Recomendaciones para el Refinamiento

Categoría	Buenas prácticas
Preparación	Llegar con backlog ordenado, foco definido y feedback analizado.
Claridad	Historias pequeñas, comprensibles y con criterios claros.
Participación	Todos opinan, no solo el Product Owner.
Priorización	Elegir por valor al usuario, no solo por facilidad técnica.
Estimación	Realista, colaborativa y guiada por la experiencia.
Seguimiento	Dejar evidencia (minuta y backlog actualizado).

Requisitos Sprint 2 – Practica I

6. Anexos del informe

3) Plantilla de Pruebas Unitarias por HU

Responsable: Tester asignado (PO o SM). Adjuntar resultados como comentario en la HU de Taiga.

HU	Caso de prueba	Entrada	Resultado esperado	Resultado obtenido	Estado (Pass/Fail)	Evidencia (link/captura)
HU-01	Entrada válida	payload JSON con campos X,Y	Respuesta 200 con campo Z	200 con Z correcto	Pass	Captura_01.png
HU-01	Campo ausente	payload sin X	Error manejado y log	Error 400 y log en n8n	Pass	Captura_02.png
HU-01	Timeout externo	simular API lenta	Reintento y alerta	Reintento x3, alerta Slack	Fail	Captura_03.png

Notas de pruebas unitarias:

- Definir al menos 3 casos por HU (válido, inválido, borde/timeout).
- Registrar versión del flujo n8n probada y las credenciales/entorno empleados.
- Guardar export del flujo (.json) con nombre estándar: HU-XX_v1.json.

Requisitos Sprint 2 – Practica I

4) Plan de Pruebas de Integración (Flujo Integrado)

Responsable: Developer Líder + Tester (PO/SM). Subir a la Wiki del Sprint y vincular a cada HU.

ID	Escenario	Precondiciones	Pasos	Resultado esperado	Evidencia
INT-01	Flujo HU1→HU2→HU3 con datos válidos	n8n listo, credenciales configuradas	Disparar HU1 con payload estándar	Ejecución completa sin errores	Video o capturas
INT-02	Falla en HU2 (API externa)	Mock activo para API	Simular error 500 en HU2	HU3 no dispara; registro de error y alerta	Logs + captura
INT-03	Datos límite	Dataset con campos extremos	Ejecutar con valores máximos/mínimos	Validaciones correctas; tiempos en rango	Reporte tiempos

Evidencia mínima del flujo integrado:

- Capturas del flujo final en n8n (diagrama y nodos clave).
- Export del flujo integrado (.json) y hash o fecha de versión.
- Logs de ejecución (OK y con fallos simulados).
- Video corto de la ejecución end-to-end (puede reutilizarse para la Review).

Requisitos Sprint 2 – Practica I

5) Nomenclatura y consistencia de artefactos

- HU y tareas: HU-XX, T-XX_HU-XX.
- Evidencia: HU-XX_TestCase-YY.png, INT-XX_Log.txt, FlujoIntegrado_v1.json.
- Video: Review_Sprint1_EquipoX.mp4.
- Ubicación en Taiga: HU → Adjuntos; Wiki Sprint → Actas e informes; Tareas → progreso.

6) Guías para mejorar las HU y la comprensión de los requisitos del proyecto

A. Clarificando los requisitos. Para cada HU debe agregar y/o revisar los siguientes anexos

1) Backlog de Preguntas al PO que explican el supuesto

ID	Prior.	HU	Pregunta	Contexto (qué probamos)	Propuesta del equipo	Impacto	Estado
Q-03	Alta	HU-02	¿Formato definitivo de fecha?	API X devuelve YYYY-MM-DD	Usar ISO 8601	Bloquea validación	Pendiente

Regla: cada pregunta debe traer **una propuesta**; si no, no se publica.

2) “Ficha de HU” (1 página por HU, estándar)

- **Objetivo de la HU** (1 frase).
- **CA (≥2) en Given/When/Then.**
- **Entradas/Salidas esperadas** (JSON breve).
- **Errores esperados & manejo.**
- **Datos de prueba** (válido/inválido/borde).
- **Evidencia** (captura/log/etc).

Requisitos Sprint 2 – Practica I

B. Guía rápida para escribir CA (pegar en Taiga en Wiki): 5 reglas simples

1. **Un comportamiento por CA.**
2. **Medible y observable** (qué devuelve/qué cambia).
3. **Incluye un CA negativo** (error o dato inválido).
4. **Datos concretos** (valores reales, no “X/Y”).
5. **Tiempo/umbral** cuando aplique (p. ej., ≤ 15 s)

Plantilla ejemplo de plantilla

- **Given** un payload con {email: válido}
- **When** se ejecuta el nodo **API_EnviarCorreo**
- **Then** el servicio responde **200** y se registra log “correo_enviado:true”
- **And** ante email inválido devuelve **400** y log de validación.

C. Definition of Ready (DoR) ligero para el refinamiento. Esto debe hacerlo el tester por HU asignada

Antes de entrar una HU al sprint:

- ☐ Objetivo claro en 1 frase.
- ☐ **≥ 2 AC** en Given/When/Then.
- ☐ Entradas/Salidas definidas (campo, tipo, ejemplo).
- ☐ Manejo de errores mínimo descrito.
- ☐ Datos de prueba listos.
- ☐ Dependencias/impacto en flujo integrado identificadas.
- ☐ Estimada (Planning Poker)

Requisitos Sprint 2 – Practica I

D. Rutina de comunicación (enseñar a “preguntar bien”. Pegar en Taiga). Anexo para explicar cómo desarrollar buenas preguntas de los requisitos

Formato de comentario en Taiga (obligatorio):

- **Contexto:** qué HU/tarea, qué intenté, qué ocurrió.
- **Evidencia:** captura/log/caso.
- **Hipótesis:** qué creo que pasa.
- **Propuesta:** A) ____ B) ____.
- **Etiqueta:** #blocking / #data / #ux / #api.