

Taller de proyectos 2 – ingeniería de Sistemas e informática

CONSIGNA ELABORACIÓN DE LA ITERACIÓN CERO

1. Objetivo

- 1.1. Diseñar y documentar la Iteración 0 de un proyecto web con MERN bajo un enfoque ágil.
- 1.2. Elaborar el Project Charter como documento base del proyecto, estableciendo el marco inicial de trabajo.

2. Consigna

En equipos, elaboren la Iteración 0 de su proyecto web considerando los siguientes entregables:

- 2.1. Project Charter (Acta de Proyecto)

PROJECT CHARTER

Propósito o justificación del proyecto

El proyecto busca mejorar las habilidades de pensamiento crítico y comprensión lectora en estudiantes y usuarios en general, a través de un tutor virtual que aprovecha Inteligencia Artificial y flujos de automatización. La iniciativa responde a la necesidad de fomentar la lectura crítica en entornos educativos y de autoaprendizaje, ofreciendo una herramienta accesible, automatizada y escalable.

Descripción general del proyecto

El proyecto consiste en el diseño e implementación de una aplicación de **tutoría virtual en lectura crítica**, que:

1. Permite a los estudiantes leer textos y responder preguntas generadas automáticamente.
2. Detecta sesgos y falacias lógicas en artículos mediante modelos de NLP.
3. Automatiza el flujo de asignación, recordatorios y registro de progreso con n8n.
4. Proporciona a docentes un espacio para asignar textos y evaluar el progreso.
5. Ofrece a administradores un panel con métricas globales de comprensión.

Requisitos de alto nivel

- Generación automática de preguntas mediante **NLP (Hugging Face u otra API ML)**.
- Clasificación de sesgos en textos (modelo entrenado o servicio externo).

- Flujo de trabajo automatizado con **n8n**: asignación → lectura → actividad → evaluación → notificación.
- Panel de control con métricas de progreso y comprensión promedio.
- Notificaciones y recordatorios para mantener hábitos de lectura.

Objetivos del proyecto y criterios de éxito

- Desarrollar un tutor virtual funcional en un plazo máximo de **6 meses**.
- Mantener el presupuesto del proyecto dentro de ±10% del monto aprobado.
- Lograr que al menos el **70% de los estudiantes** mejoren su comprensión crítica en pruebas piloto.
- Implementar un flujo estable de notificaciones y registro de respuestas vía n8n.
- Obtener la validación positiva de docentes y usuarios en pruebas de usabilidad.

Riesgos de alto nivel

- Complejidad en el entrenamiento y desempeño del modelo de detección de sesgos.
- Integración limitada entre la aplicación y n8n si existen restricciones técnicas.
- Baja adopción de la herramienta por falta de motivación de los usuarios.
- Costos adicionales asociados al uso de APIs externas de NLP.

Cronograma resumido de hitos

- **Inicio del proyecto** – Semana 1
- **Prototipo funcional de preguntas automáticas** – Semana 4
- **Integración con n8n para notificaciones y registro** – Semana 8
- **Desarrollo del panel de administración** – Semana 12
- **Prueba piloto con estudiantes y docentes + Implementación final y despliegue** – Semana 14

Presupuesto resumido

Presupuesto estimado: **\$67,044** (incluye desarrollo, integración con n8n, pruebas piloto, entrenamiento de modelos y soporte inicial).

Principales interesados (stakeholders)

- **Gerente de proyecto:** Julio Cesar Quispe Zarate

- **Equipo de desarrollo:** programadores, expertos en NLP, especialistas en automatización.
- **Docentes:** usuarios clave para asignación de textos y evaluación.
- **Estudiantes / usuarios finales:** beneficiarios principales.
- **Administradores:** responsables del monitoreo del desempeño del sistema.

Requisitos de aprobación del proyecto

El proyecto será considerado finalizado y aprobado si:

1. La aplicación está en producción y disponible para los usuarios.
2. Los flujos automatizados con n8n funcionan de extremo a extremo.
3. El panel de métricas está operativo y refleja datos en tiempo real.
4. Al menos un 70% de los usuarios piloto reportan mejoras en comprensión crítica.

Gerente de proyecto asignado, responsabilidad y nivel de autoridad

Nombre: Julio Cesar Quispe Zarate

Autoridad:

- Puede gestionar al equipo de desarrollo y coordinar con docentes y administradores.
- Tiene autoridad sobre la asignación de recursos y control del cronograma.
- No tiene autoridad para aprobar incrementos presupuestales mayores al 10% sin el visto bueno del equipo de desarrollo.

2.1.1. Propósito y justificación del proyecto.

Para estudiantes y docentes que buscan fortalecer habilidades de análisis crítico, el Tutor Virtual de Lectura Crítica es una aplicación avanzada que integra inteligencia artificial para generar preguntas, detectar sesgos y evaluar la comprensión del texto. A diferencia de otros métodos tradicionales, este sistema automatiza la asignación de actividades, el seguimiento de progreso y la detección de fallas lógicas, con un enfoque en la mejora continua del usuario mediante recordatorios y hábitos diarios.

2.1.2. Objetivos generales.

Trabajar de manera colaborativa y eficiente para entregar la aplicación del Tutor Virtual de Lectura Crítica dentro del plazo establecido, asegurando que cumpla con las expectativas de calidad, funcionalidad y usabilidad. El objetivo es facilitar el desarrollo de habilidades de pensamiento crítico a través de una herramienta intuitiva y automatizada.

2.1.3. Alcance inicial.

- Desarrollo de la plataforma web/móvil para usuarios finales.
- Integración de modelos de IA para generación de preguntas y detección de sesgos.
- Automatización con n8n para programación de sesiones y seguimiento.
- Visualización de métricas de comprensión para administradores.

2.1.4. Supuestos y restricciones.

ID	Categoría	Supuesto/Restricción	Estado	Comentarios
A1	Habilidades	Suponemos que hay disponibilidad de expertos en NLP e IA para desarrollar y entrenar los modelos.	Pendiente	Se cuenta con profesionales, pero depende de la contratación y disponibilidad a tiempo completo.
A2	Recursos	Suponemos que la plataforma n8n estará disponible y se podrá integrar sin problemas en el sistema.	Confirmado	n8n es open source y ampliamente usado; se ha verificado compatibilidad preliminar.
A3	Usuarios	Suponemos que los estudiantes y docentes adoptarán el uso del tutor virtual como parte de sus rutinas.	Pendiente	Será necesario impulsar la capacitación y difusión para asegurar la adopción.
C1	Técnico	Restricción: La aplicación debe cumplir con las políticas de privacidad y protección de datos estudiantiles (ej. GDPR o equivalente local).	Activo	Implica asegurar seguridad en el manejo de datos y consentimientos explícitos.
C2	Tiempo	Restricción: El proyecto debe estar finalizado y en operación en un plazo máximo de 9 meses.	Activo	Se debe ajustar el cronograma para cumplir con esta fecha límite establecida por la dirección.

2.1.5. Identificación de los interesados (stakeholders).

Principales interesados (stakeholders)

- **Gerente de proyecto:** Julio Cesar Quispe Zarate
- **Equipo de desarrollo:** programadores, expertos en NLP, especialistas en automatización.
- **Docentes:** usuarios clave para asignación de textos y evaluación.

- **Estudiantes / usuarios finales:** beneficiarios principales.
- **Administradores:** responsables del monitoreo del desempeño del sistema.

2.1.6. Roles y responsabilidades del equipo.

1. Miembros y roles

Nombre	Roles	Responsabilidades claves
Auccatoma Rojas Jack Jeysen	Project Manager	Liderar el proyecto, gestionando cronograma, presupuesto, riesgos, y comunicación con los stakeholders para asegurar el cumplimiento de los objetivos.
Chambi Rodriguez Fernando Meliton	Product Owner	Definir la visión del producto, priorizar el backlog, tomar decisiones clave y garantizar que las funcionalidades sean claras para el equipo de desarrollo.
Quispe Zarate Cesar	Scrum Master	Facilitar las ceremonias de Scrum, resolver obstáculos, promover la auto-organización del equipo y mantener la comunicación entre el equipo y el Product Owner.
Olivares Collachagua Juldant Walter	Scrum Master	
López Cañete Sebastián Mariano	Equipo de desarrollo: Backend y Frontend	Desarrollar la interfaz de usuario (UI), integrar funcionalidades del sistema, asegurar la responsividad y realizar pruebas de usabilidad para una experiencia fluida.
Caballero Capcha Maruja Mistica	Equipo de desarrollo: Backend y Frontend	

2.2. Backlog inicial

ID	Épica	Historia de Usuario	Prioridad (MoSCoW)	Criterios de Aceptación
E1	Tutor de Lectura Crítica	Como estudiante, quiero responder preguntas automáticas sobre el texto leído para mejorar mi pensamiento crítico.	M	El sistema genera al menos 5 preguntas por texto; las preguntas son relevantes al contenido.

E1 -2	Tutor de Lectura Crítica	Como estudiante, quiero recibir retroalimentación inmediata sobre mis respuestas para entender mis aciertos y errores.	M	El sistema muestra retroalimentación clara (correcto/incorrecto + explicación).
E1 -3	Tutor de Lectura Crítica	Como estudiante, quiero guardar mi historial de respuestas para revisar mi progreso.	S	El sistema almacena respuestas y muestra reportes de progreso por fecha.
E1 -4	Tutor de Lectura Crítica	Como estudiante, quiero recibir sugerencias de lectura complementaria para profundizar en el análisis.	C	El sistema recomienda al menos un texto adicional relacionado al tema.
E2	Detección de Sesgos y Comprensión	Como usuario, quiero que el sistema detecte sesgos o falacias lógicas en los textos para mejorar mi capacidad crítica.	M	El sistema resalta pasajes con posibles sesgos o falacias y los clasifica.
E2 -2	Detección de Sesgos y Comprensión	Como estudiante, quiero visualizar ejemplos de sesgos detectados en distintos textos para aprender a reconocerlos.	S	El sistema muestra ejemplos etiquetados y explicados.
E2 -3	Detección de Sesgos y Comprensión	Como docente, quiero recibir un reporte automático con los sesgos detectados en los textos asignados a mis estudiantes.	S	El sistema genera reporte consolidado por clase.
E2 -4	Detección de Sesgos y Comprensión	Como usuario, quiero poder marcar manualmente un sesgo que el sistema no detectó para mejorar su precisión.	C	El sistema permite etiquetas manuales y feedback.
E3	Gestión y Automatización de Aprendizaje (n8n)	Como docente, quiero asignar textos automáticamente con actividades para que mis estudiantes	M	El flujo automatiza: asignación → lectura → actividad.

		trabajen sin intervención manual.		
E3 -2	Gestión y Automatización de Aprendizaje (n8n)	Como estudiante, quiero recibir notificaciones de recordatorio para mantener la constancia en mis sesiones de lectura crítica.	M	El sistema envía notificaciones diarias/semanales configurables.
E3 -3	Gestión y Automatización de Aprendizaje (n8n)	Como administrador, quiero visualizar un panel con el nivel de comprensión promedio de los usuarios para monitorear resultados.	M	Dashboard muestra métricas agregadas (promedios, tendencias).
E3 -4	Gestión y Automatización de Aprendizaje (n8n)	Como estudiante, quiero configurar mis horarios preferidos para recibir los textos y recordatorios.	S	El sistema permite personalizar horarios.
E3 -5	Gestión y Automatización de Aprendizaje (n8n)	Como sistema, quiero registrar automáticamente las respuestas y enviarlas a la base de datos para consolidar hábitos de lectura.	M	Respuestas se guardan en la base de datos en tiempo real.
E3 -6	Gestión y Automatización de Aprendizaje (n8n)	Como administrador, quiero exportar datos del progreso en formato Excel o PDF para análisis externo.	C	El sistema genera reportes descargables.

Identificación del Epic central y descomposición en historias.

Épica 1: Tutor de Lectura Crítica

Historia 1.1: Como estudiante, quiero responder preguntas automáticas sobre el texto leído, para mejorar mi pensamiento crítico.

- **Prioridad:** Alta
- **Estimación:** 10 horas

Criterios de aceptación:

- El sistema genera al menos 5 preguntas relevantes por texto.
- El estudiante puede responder en formato opción múltiple o texto corto.
- Las preguntas están alineadas al contenido del texto.

Historia 1.2: Como estudiante, quiero recibir retroalimentación inmediata sobre mis respuestas, para comprender mis aciertos y errores.

- **Prioridad:** Alta
- **Estimación:** 8 horas

Criterios de aceptación:

- Después de responder, el sistema muestra “Correcto/Incorrecto”.
- Se ofrece una explicación breve o pista adicional.
- Los resultados quedan guardados en el perfil del estudiante.

Historia 1.3: Como estudiante, quiero guardar mi historial de respuestas, para revisar mi progreso a lo largo del tiempo.

- **Prioridad:** Media
- **Estimación:** 6 horas

Criterios de aceptación:

- El usuario puede acceder a un historial con fecha, texto y puntaje.
- El progreso se muestra en gráficos o tablas.

Historia 1.4: Como estudiante, quiero recibir sugerencias de lecturas complementarias, para profundizar mi análisis crítico.

- **Prioridad:** Baja
- **Estimación:** 6 horas

Criterios de aceptación:

- El sistema recomienda al menos un texto relacionado.
- Las sugerencias se generan en base al tema leído o a nivel de desempeño.

Épica 2: Detección de Sesgos y Comprensión

Historia 2.1: Como usuario, quiero que el sistema detecte sesgos o falacias lógicas en los textos, para mejorar mi

capacidad crítica.

- **Prioridad:** Alta
- **Estimación:** 12 horas

Criterios de aceptación:

- El sistema subraya fragmentos sospechosos de sesgos.
- Clasifica el tipo de sesgo (ej. falacia ad hominem, generalización).

Historia 2.2: Como estudiante, quiero visualizar ejemplos de sesgos detectados en distintos textos, para aprender a reconocerlos.

- **Prioridad:** Media
- **Estimación:** 8 horas

Criterios de aceptación:

- Se muestra un repositorio con textos anotados y explicación de los sesgos.
- El estudiante puede acceder como práctica adicional.

Historia 2.3: Como docente, quiero recibir un reporte automático con los sesgos detectados en los textos asignados, para evaluar cómo mis estudiantes los identifican.

- **Prioridad:** Media
- **Estimación:** 10 horas

Criterios de aceptación:

- El sistema genera un reporte consolidado en PDF o Excel.
- Incluye estadísticas del número y tipo de sesgos detectados.

Historia 2.4: Como usuario, quiero poder marcar manualmente un sesgo no detectado, para retroalimentar y mejorar el sistema.

- **Prioridad:** Baja
- **Estimación:** 6 horas

Criterios de aceptación:

- El usuario puede seleccionar un fragmento y etiquetarlo como sesgo.
- El sistema registra el feedback para mejorar el modelo.

Épica 3: Gestión y Automatización con n8n

Historia 3.1: Como docente, quiero asignar textos automáticamente con actividades, para que mis estudiantes trabajen sin intervención manual.

- **Prioridad:** Alta
- **Estimación:** 12 horas

Criterios de aceptación:

- El docente selecciona texto y fecha.
- El flujo n8n programa la actividad y la envía automáticamente.

Historia 3.2: Como estudiante, quiero recibir notificaciones de recordatorio, para mantener la constancia en mis sesiones de lectura.

- **Prioridad:** Alta
- **Estimación:** 8 horas

Criterios de aceptación:

- El sistema envía recordatorios diarios o semanales configurables.
- Notificaciones llegan vía correo o aplicación móvil.

Historia 3.3: Como administrador, quiero visualizar un panel con el nivel de comprensión promedio de los usuarios, para monitorear el desempeño global.

- **Prioridad:** Alta
- **Estimación:** 10 horas

Criterios de aceptación:

- El dashboard muestra métricas agregadas (promedio de puntajes, tendencias).
- Datos filtrables por rango de fechas y grupo de usuarios.

Historia 3.4: Como estudiante, quiero configurar mis horarios preferidos para recibir los textos y recordatorios, para ajustar la aplicación a mi rutina.

- **Prioridad:** Media
- **Estimación:** 6 horas

Criterios de aceptación:

- El usuario puede elegir horarios desde la configuración.
- El sistema respeta esas preferencias al enviar recordatorios.

Historia 3.5: Como sistema, quiero registrar automáticamente las respuestas en la base de datos, para consolidar hábitos de lectura.

- **Prioridad:** Alta
- **Estimación:** 6 horas

Criterios de aceptación:

- Todas las respuestas se almacenan con fecha y hora.
- Los datos quedan disponibles para reportes y métricas.

Historia 3.6: Como administrador, quiero exportar datos del progreso en formato Excel o PDF, para análisis externo.

- **Prioridad:** Baja
- **Estimación:** 6 horas

• Criterios de aceptación:

- El panel tiene botón de exportación.
- Archivos contienen métricas individuales y agregadas.

2.2.2. Lista de 6 a 8 historias de usuario priorizadas con INVEST.

Resumen de Prioridades

- **Alta (Core inicial):** 1.1, 1.2, 2.1, 3.1, 3.2, 3.3, 3.5
- **Media (Segunda iteración):** 1.3, 2.2, 2.3, 3.4
- **Baja (Mejoras futuras):** 1.4, 2.4, 3.6

2.3. Infraestructura y entorno técnico

2.3.1. Creación de repositorio en GitHub (un integrante crea, el resto clona).

2.3.2. Configuración inicial de ramas (main, develop, ramas por funcionalidad).

2.3.3. Estructura base del proyecto MERN (carpetas, dependencias mínimas).

2.4. Procesos de equipo

2.4.1. Definición de canales de comunicación (ej. Slack, Discord).

Herramientas de comunicación y gestión:

Herramienta	Uso Principal
Slack / WhatsApp	Coordinación diaria, avisos rápidos y comunicación instantánea.
Trello / Jira	Gestión de backlog, asignación de tareas y control de progreso.
Google Meet / Zoom	Reuniones virtuales, planificación y revisiones.
GitHub / GitLab	Control de versiones, colaboración en código y documentación técnica.

2.4.2. Acuerdos de trabajo: horarios de reunión, Definition of Done, roles iniciales.

Reunión	Frecuencia / Duración	Propósito Principal
Daily Stand-up	Diario – 15 min	Seguimiento de tareas, identificar impedimentos y alinear al equipo.
Sprint Planning	Inicio de cada sprint – 1-2 hrs	Definir objetivos del sprint y priorizar el backlog.
Sprint Review	Fin de cada sprint – 1-2 hrs	Presentar avances al Product Owner y patrocinador; recibir feedback.
Sprint Retrospective	Fin de cada sprint – 1 hr	Analizar fortalezas y debilidades del proceso; proponer mejoras.

2.5. Arquitectura mínima (Walking Skeleton)

2.5.1. Configuración básica: conexión Express–MongoDB, Hello World en React.

2.5.2. Evidencia de despliegue inicial en un servicio gratuito (ej. Render, Vercel, Railway).

2.6. Entregables

2.6.1. Documento (Markdown, PDF o Google Docs) con:

- Project Charter completo.
- Backlog inicial y acuerdos de equipo.

2.6.2. URL del repositorio GitHub con:

- Código base MERN configurado.
- Readme con instrucciones de instalación.

2.6.3. Evidencia del despliegue inicial (screenshot o URL).

The screenshot shows a Jira backlog board for the project 'Proyecto Revisión de textos2'. The board has two main sections: 'Tablero Sprint 1' and 'Backlog'.

Tablero Sprint 1: Contains 7 activities. Each activity includes a summary, assignee, status, and estimated effort (TAREAS POR ...). The activities are:

- PRDT2-1 HU1:Cuestionario preguntas automáticas (EP1:TUTOR ...)
- PRDT2-3 HU2:Retroalimentacion cuestionario (EP1:TUTOR ...)
- PRDT2-4 HU3:Guardas historial de respuestas (EP1:TUTOR ...)
- PRDT2-10 HU9:Asignar textos automáticamente a Estudiantes (EP3:GESTIÓN...)
- PRDT2-11 HU10:Recibir notificaciones para constancia lectura critica (EP3:GESTIÓN...)
- PRDT2-12 HU11:Panel con nivel de comprensión usuarios (EP3:GESTIÓN...)
- PRDT2-14 HU13:Registrar respuestas para BD (EP3:GESTIÓN...)

Backlog: Contains 7 activities. Each activity includes a summary, assignee, status, and estimated effort (TAREAS POR ...). The activities are:

- PRDT2-5 HU4:Recibir Sugerencia de lectura (EP1:TUTOR ...)
- PRDT2-6 HU5:Detectar Sesgos o Falacias (EP2:DETECC...)
- PRDT2-7 HU6:Visualizar Ejemplos de Sesgo (EP2:DETECC...)
- PRDT2-8 HU7:Reporte de sesgos Estudiantes (EP2:DETECC...)
- PRDT2-9 HU8:Señalar Sesgo manualmente (EP2:DETECC...)
- PRDT2-13 HU12:Configurar horarios como estudiante (EP3:GESTIÓN...)
- PRDT2-15 HU14:Exportar datos como Excel o PDF (EP3:GESTIÓN...)

The screenshot shows a Jira backlog board for the project 'Proyecto Revisión de textos2'. The board has two main sections: 'Backlog' and 'Backlog' (7 actividades).

Backlog: Contains 7 activities. Each activity includes a summary, assignee, status, and estimated effort (TAREAS POR ...). The activities are:

- PRDT2-5 HU4:Recibir Sugerencia de lectura (EP1:TUTOR ...)
- PRDT2-6 HU5:Detectar Sesgos o Falacias (EP2:DETECC...)
- PRDT2-7 HU6:Visualizar Ejemplos de Sesgo (EP2:DETECC...)
- PRDT2-8 HU7:Reporte de sesgos Estudiantes (EP2:DETECC...)
- PRDT2-9 HU8:Señalar Sesgo manualmente (EP2:DETECC...)
- PRDT2-13 HU12:Configurar horarios como estudiante (EP3:GESTIÓN...)
- PRDT2-15 HU14:Exportar datos como Excel o PDF (EP3:GESTIÓN...)

Resumen Cronograma Backlog Sprints activos Calendario Informes Lista Formularios Metas More 7 +

Buscar en el crono... jc Epic Categoría de es... Filtros rápidos

Actividad	September	October	November
Sprints	Tablero Sprint 1		
Publicaciones		version pmv	
<input type="checkbox"/> PRDT-1 EP01			
<input type="checkbox"/> PRDT-2 EP02			
<input type="checkbox"/> PRDT-3 EP03			
+ Crear Epic			

Taller de proyectos 2 – ingeniería de Sistemas e informática

RÚBRICA DE EVALUACIÓN – ITERACIÓN 0 / INCEPTION SPRINT

Criterio / Indicador	Sobresaliente (3)	Suficiente (2)	En desarrollo (1)	Insatisfactorio (0)
Project Charter: Contiene propósito, objetivos, alcance, supuestos, restricciones, stakeholders y roles; la redacción es clara y coherente.	Documentación completa, clara y bien estructurada, todos los apartados están presentes y se incluyen aportes adicionales.	Incluye todos los apartados mínimos solicitados con redacción aceptable y sin errores críticos.	El documento está incompleto o algunos apartados son vagos o mal desarrollados.	No se presenta el documento o tiene errores graves en la mayoría de apartados.
Backlog inicial: Incluye al menos 6–8 historias de usuario con INVEST, un épico central desglosado y una priorización clara.	Todas las historias cumplen con INVEST, se incluyen criterios de aceptación claros y la priorización está justificada con valor de negocio.	Historias redactadas con INVEST básico, cumplen con la cantidad mínima y presentan orden lógico.	Menos de 6 historias o varias mal redactadas; sin criterios de aceptación ni priorización clara.	No hay backlog o las historias no cumplen con la estructura mínima.
Organización y acuerdos de equipo: Se definen canales de comunicación, roles iniciales, horarios de reunión y una Definition of Done consensuada.	Se detallan acuerdos claros, realistas y completos, incluyen métricas de calidad y herramientas de seguimiento; todos los integrantes participan en la definición.	Se documentan acuerdos mínimos (canales, roles, horarios y DoD) de manera clara.	Los acuerdos son parciales, vagos o sin consenso; la DoD es incompleta.	No hay acuerdos documentados o son incoherentes con la dinámica ágil.
Repositorio y estructura técnica: Se crea repositorio en GitHub, con ramas organizadas, estructura base MERN y README con instrucciones claras.	Repositorio con ramas bien gestionadas, commits descriptivos, estructura MERN completa, README con instalación y contribución; integración inicial verificada.	Repositorio creado, ramas mínimas (<i>main, develop</i>), estructura básica MERN y README con instrucciones de instalación.	Repositorio desordenado, sin ramas claras, con errores en estructura o README incompleto.	No hay repositorio o la entrega es inconsistente con la consigna.

Arquitectura mínima (Walking Skeleton): Implementación básica MERN con conexión Express– MongoDB, Hello World en React y despliegue en la nube con evidencia.	Proyecto ejecuta correctamente con Express–MongoDB conectados, frontend React desplegado, despliegue documentado con URL y pruebas básicas incluidas.	Proyecto con Express y React configurados, base de datos conectada, despliegue funcional mínimo con URL accesible.	Implementación parcial (ej. solo backend o solo frontend), errores en conexión o despliegue incompleto.	No se presenta un esqueleto funcional o no hay evidencia de despliegue.
Presentación Oral (Equipo): Claridad, organización y	Presentación profesional, engaging y bien coordinada; todos los	Presentación clara y organizada; la mayoría de los miembros	Presentación desorganizada o confusa; participación desigual	La presentación no se realiza o es extremadamente

Taller de proyectos 2 – ingeniería de Sistemas e informática

Criterio / Indicador	Sobresaliente (3)	Suficiente (2)	En desarrollo (1)	Insatisfactorio (0)
participación equitativa de todos los miembros; capacidad de síntesis y respuesta a preguntas.	miembros participan activamente; explican con claridad los documentos y responden preguntas con solidez.	participan; logran comunicar los puntos principales, pero con falta de fluidez o profundidad en las respuestas.	entre miembros; dificultad para explicar el contenido de los documentos o responder preguntas.	deficiente, sin lograr comunicar el lanzamiento del proyecto.

Formalidad y Entrega: Correcta estructuración y redacción de los 4 documentos en PDF; entrega puntual por el aula virtual.	Los 4 documentos están redactados con excelente formalidad académica, sin errores gramaticales; estructura clara y profesional; entregados a tiempo y en el formato correcto.	Los 4 documentos están redactados adecuadamente, con algunos errores formales menores; entregados a tiempo y en el formato solicitado.	Documentos con redacción informal, múltiples errores o estructura deficiente; o entrega fuera de plazo o en formato incorrecto.	No se entregan los documentos o se entrega una cantidad menor a los 4 requeridos.
-----------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------