



PROCESO		
MÓDULO		
NOMBRE DEL FORMATO		
INFORME DE ACTIVIDADES SEMANALES		
CLASIFICACIÓN DE LA INFORMACIÓN		
Pública ☐	Pública Clasificada ☒	Pública Reservada ☐

PROYECTO ZAJUNA SENA

SEMANA DEL 9 AL 16 DE MARZO DE 2026

Presentado por:

LUIS ANDRÉS RÍOS NOREÑA

Centro de Diseño e Innovación Tecnológica Industrial

SENA Risaralda

Dosquebradas

2026

ENTREGABLES:

GOR-F-012 V03



→ Actividad 1.

Consolidación de Documentación Técnica y Gestión de Sprint.

Esta actividad se centró en la formalización de la documentación técnica necesaria para el despliegue del proyecto Zajuna AI y el registro de estos avances en la herramienta de gestión Taiga. Se consolidó el material de instalación y despliegue (requisitos, configuración de entorno y persistencia), se documentó el procedimiento para aprovisionar modelos de lenguaje desde Hugging Face, y se dejó trazabilidad de los artefactos y estados de validación asociados a la época de documentación.

EVIDENCIAS DEL ENTREGABLE

A continuación, se puede observar las pruebas referentes a los entregables:

- Anexar los 3 Entregables

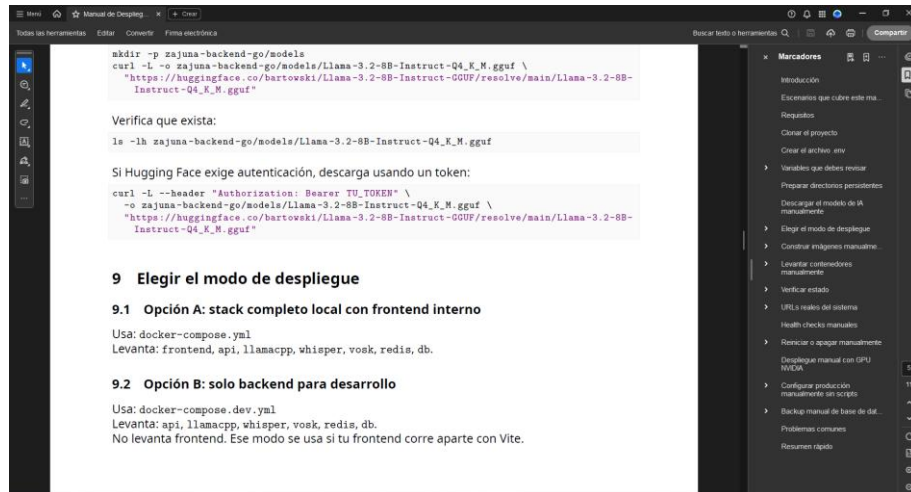
Evidencia 1 (Manual de Despliegue): Se visualiza la portada y tabla de contenidos del “Manual de Despliegue de Zajuna AI”, el cual incluye secciones críticas como requisitos, clonación del proyecto, configuración de variables de entorno (.env) y preparación de directorios persistentes.

Manual de Despliegue Manual de Zajuna AI	
Documentación Técnica	
Tabla de Contenidos	
1	Introducción 3
2	Escenarios que cubre este manual 3
3	Requisitos 3
4	Clonar el proyecto 4
5	Crear el archivo .env 4
6	Variables que debes revisar 4
6.1	Obligatorias para producción 4
6.2	URLs del frontend 4
6.3	Puertos 4

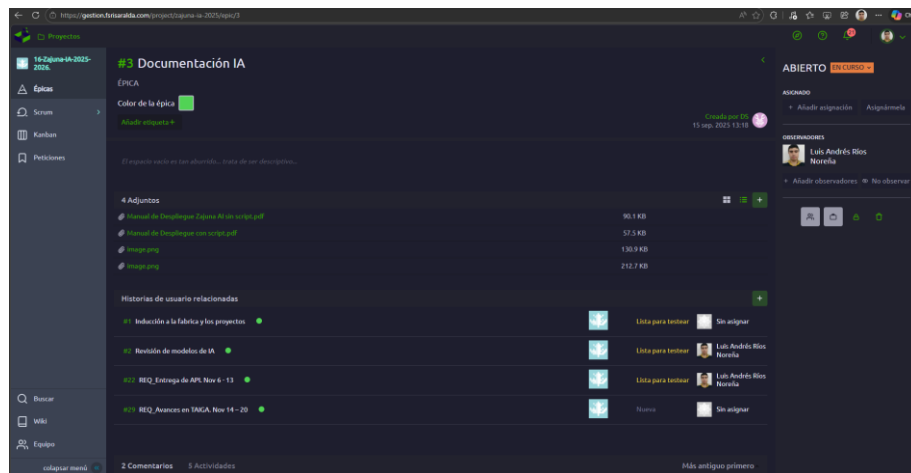
Evidencia 2 (Instrucciones Técnicas): Captura detallada de comandos de terminal (curl) para la descarga de modelos de lenguaje (Llama-3.2-8B) desde Hugging Face y la



configuración del stack con docker-compose. Se especifican opciones de despliegue para el stack completo o únicamente el backend.



Evidencia 3 (Trazabilidad en Taiga): Interfaz de la épica “#3 Documentación IA” en Taiga, donde se listan adjuntos (manuales en PDF) y el estado de historias de usuario relacionadas, como “Revisión de modelos de IA” y “REQ_Entrega de API”, ambas marcadas como “Lista para testear”.





→ Actividad 2.

Implementación de Módulos Administrativos y Gestión de Roles.

Se trabajó en el desarrollo y validación del módulo de administración de usuarios, enfocándose en la definición, visualización y trazabilidad de permisos dentro de la plataforma. Se revisó la creación/edición de roles y sus descripciones funcionales, se validaron propiedades y contextos de actuación (Sistema, Usuario, Curso, etc.), y se documentaron en Taiga los requerimientos asociados a la épica de microservicios (endpoints autorizados y contratos de integración para el flujo UI/UX).

EVIDENCIAS DEL ENTREGABLE

A continuación, se puede observar las pruebas referentes a los entregables:

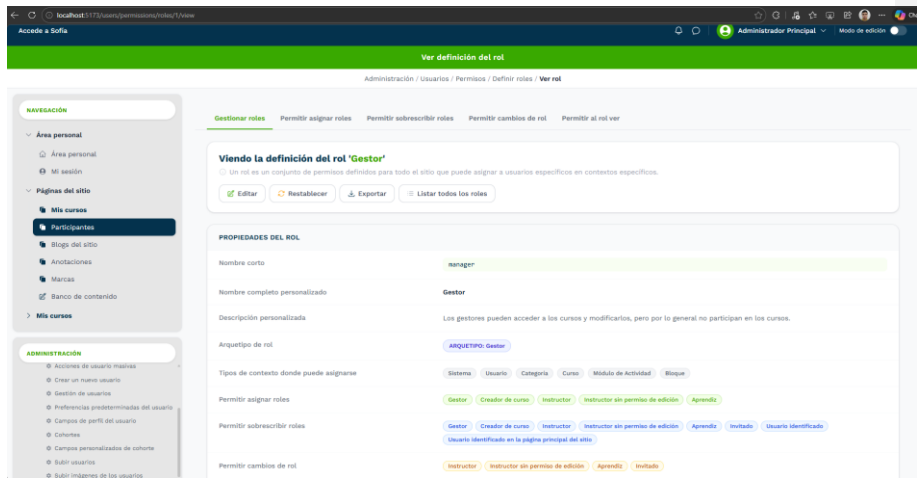
- Anexar los 3 Entregables

Evidencia 1 (Interfaz Definir Roles): Vista del panel administrativo donde se gestionan roles como Gestor, Creador de curso, Instructor y Soporte técnico. La tabla presenta el nombre corto del rol, su descripción funcional y opciones de edición/priorización.

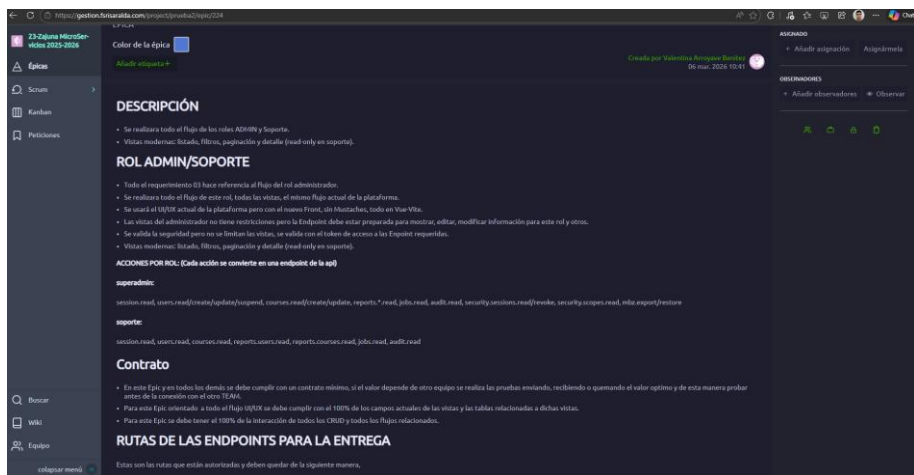
Rol	Descripción	Nombre corto	Editar
Gestor	Los gestores pueden acceder a los cursos y modificarlos, pero por lo general no participan en los cursos.	manager	↑ ↓ ⚙
Creador de curso	Los creadores de cursos pueden crear nuevos cursos.	coursecreator	↑ ↓ ⚙
Instructor	Los instructores pueden realizar cualquier acción dentro de un curso, incluyendo cambiar actividades y calificar a los aprendices.	eddingteacher	↑ ↓ ⚙
Instructor sin permisos de edición	Los instructores sin permisos de edición pueden enseñar en los cursos y calificar a los aprendices, pero no pueden modificar las actividades.	teacher	↑ ↓ ⚙
Aprendiz	Los aprendices tienen por lo general menos privilegios dentro de un curso.	student	↑ ↓ ⚙
Invitado	Los invitados tienen privilegios mínimos y normalmente no están autorizados para escribir.	guest	↑ ↓ ⚙
Usuario identificado	Todos los usuarios identificados.	user	↑ ↓ ⚙
Usuario identificado en la página principal del sitio	Todos los usuarios identificados en el curso de la página principal del sitio.	homepage	↑ ↓ ⚙
Soporte técnico	Rol de Soporte técnico.	support	↑ ↓ ⚙
Coordinador Académico	Este rol solo puede observar en la plataforma, entrar a las fichas, ver demás elementos pero no puede modificar ni cambiar nada en la plataforma.	academiccoordinator	↑ ↓ ⚙



Evidencia 2 (Detalle del Rol): Captura de la vista “Ver definición del rol ‘Gestor’”, donde se detallan propiedades del rol, el arquetipo asignado y los tipos de contexto donde puede actuar (Sistema, Usuario, Curso, etc.).



Evidencia 3 (Backlog de Microservicios): Estado de la épica “#23-Zajuna Microservicios” en la herramienta de gestión, con requerimientos para los roles ADMIN y Soporte, incluyendo rutas de endpoints autorizadas y contratos de integración para el flujo UI/UX.



→ Actividad 3.

Pruebas Unitarias y Cobertura de Código en Frontend.

Esta labor consistió en la ejecución y revisión de pruebas para asegurar la calidad del código en componentes desarrollados en Vue 3. Se generó y analizó el reporte de cobertura para identificar brechas por archivo/componente, se revisaron casos de prueba en Vitest y Vue Test Utils para validar comportamiento y eventos del frontend, y se registró la trazabilidad del avance y estado de validación de la tarea correspondiente en Taiga.

EVIDENCIAS DEL ENTREGABLE

A continuación, se puede observar las pruebas referentes a los entregables:

- Anexar los 3 Entregables.

Evidencia 1 (Reporte de Cobertura v8): Informe en consola con el resumen de cobertura (54.58% en líneas) y desglose por archivo, destacando componentes como BasePagination.vue (88.23%) y clientes de API como usersClient.ts (100%).

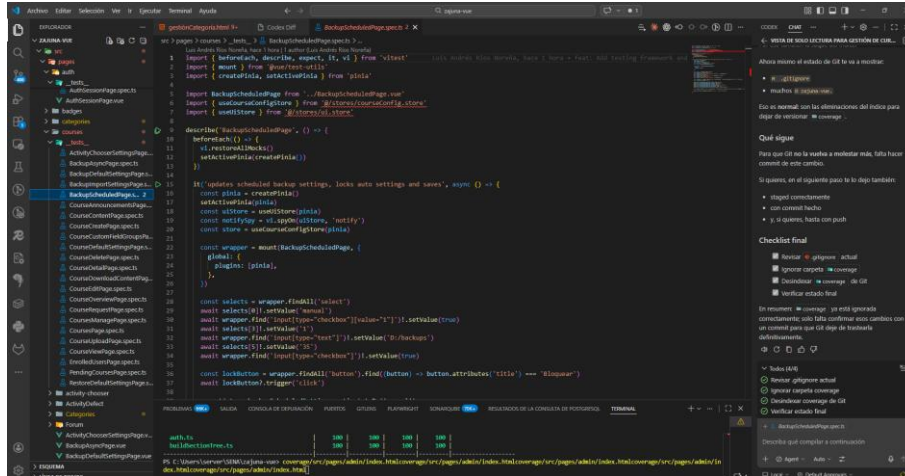


Duration 68.78s (transform 152.15s, setup 0ms, import 479.31s, tests 65.48s, environment 857.89s)

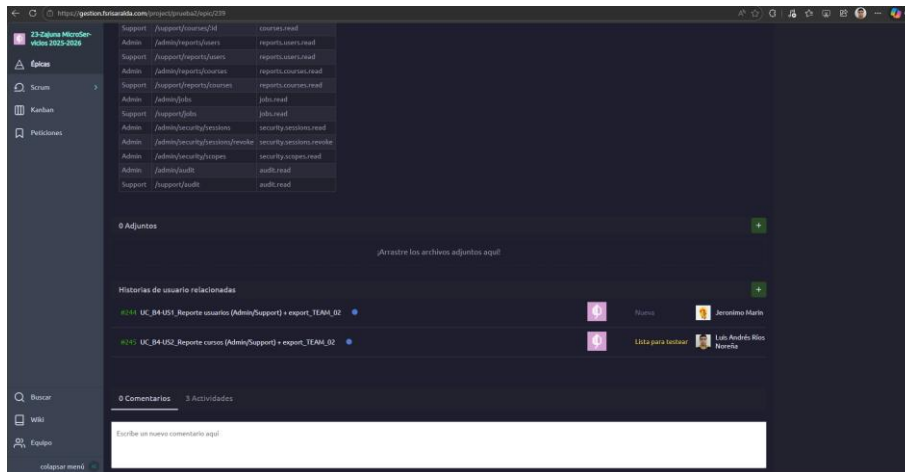
% Coverage report from v8

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	54.02	55.33	42.91	54.58	
src	0	100	0	0	
App.vue	0	100	0	0	5-13
src/_tests_	0	100	100	0	
setup.ts	0	100	100	0	3
src/api	95.83	100	75	95.83	
auditClient.ts	100	100	100	100	
cohortsClient.ts	100	100	100	100	
coursesClient.ts	100	100	100	100	
createClient.ts	92.85	100	75	92.85	35
customFieldsClient.ts	100	100	100	100	
gradesClient.ts	100	100	100	100	
securitySessionsClient.ts	100	100	100	100	
usersClient.ts	100	100	100	100	
src/auth	90	100	75	90	
permissions.ts	100	100	100	100	
sessionMemory.ts	85.71	100	75	85.71	9
src/components/common	63.86	62.4	65.9	65.06	
AppLogo.vue	0	0	100	0	4-10
BaseAlert.vue	100	100	100	100	
BaseModal.vue	100	100	100	100	
BasePagination.vue	90	91.3	77.77	88.23	37,55
BaseSelect.vue	91.42	96.66	85.71	96.77	41
BaseSpinner.vue	100	100	100	100	
BaseTable.vue	45.94	30	33.33	48.48	40-67,84,125
Configzajuna.vue	89.28	100	66.66	89.28	47,78,103
ConfirmDialog.vue	0	0	0	0	9-36
FilterPanel.vue	84.61	72.41	100	86.36	52-55
FormTooltip.vue	0	0	0	0	6-39
ImportFile.vue	84.37	90.47	75	92.85	55,94
ModalEliminar.vue	0	0	0	0	10-39
TagInput.vue	95	95	87.5	94.44	43
UserMat.vue	0	0	0	0	3-202
UserProfilePageLayout.vue	0	0	100	0	14-22
details_Mat.vue	0	0	0	0	3-114
src/components/courses	12.5	2.38	5.55	12.66	
BackupFileUploader.vue	0	0	0	0	4-151
BackupFilesTable.vue	0	0	0	0	5-101
CategoryActionsMenu.vue	0	0	0	0	9-54
CourseCard.vue	100	100	100	100	
CourseForm.vue	0	0	0	0	24-157
CourseListItem.vue	55.55	0	0	62.5	26-42

Evidencia 2 (Entorno de Desarrollo): Captura de Visual Studio Code mostrando el archivo de pruebas BackupScheduledPage.spec.ts y el uso de Vitest junto con Vue Test Utils para simular el montaje de componentes y validar eventos del frontend.



Evidencia 3 (Validación en Taiga): Seguimiento de la tarea #245 “Reporte cursos (Admin/Support)” en Taiga, en estado “Lista para testear”, bajo la responsabilidad de Luis Andrés Ríos Noreña.



→ Actividad 4.

Seguimiento de Planificación Semanal.

Comentado [HS1]: Buenas tardes, Durante la ejecución de la actividad es necesario que se anexe evidencia del entregable (en construcción o finalizado), de lo contrario se asume como una actividad que no se encuentra en desarrollo durante la semana equivalente al informe.

8

GOR-F-012 V03

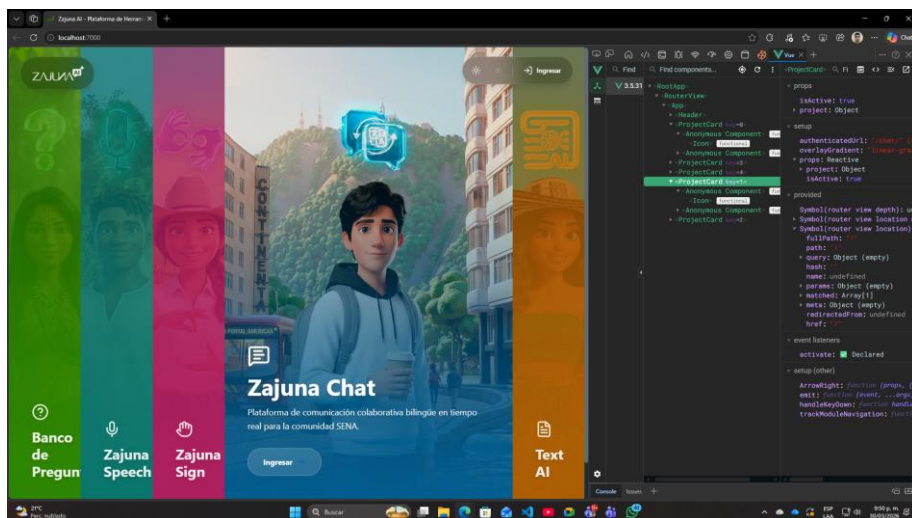


La landing fue desarrollada en Vue tras la migración desde React, ya que Vite ofrece una integración nativa y optimizada con este framework, lo que permite un mejor rendimiento en desarrollo, una configuración más simple y un Hot Module Replacement más rápido y estable; además, Vue facilita una arquitectura más clara y mantenible mediante sus Single File Components, reduce el tamaño del bundle final y evita la sobrecarga de configuración que React requiere para funcionar eficientemente con Vite, siendo una opción más adecuada para una landing page moderna y ligera.

EVIDENCIAS DEL ENTREGABLE

A continuación, se puede observar las pruebas referentes a los entregables:

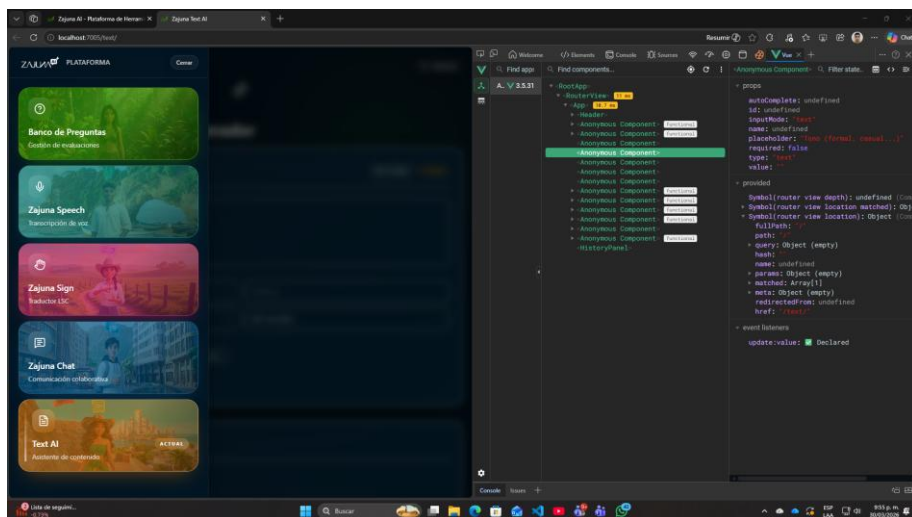
Evidencia 1 (Interfaz Principal de Zajuna AI): La imagen es una captura de pantalla dividida en dos partes. En el lado izquierdo, se muestra la *landing page* de la plataforma Zajuna AI ejecutándose en un entorno local (localhost:7000). Se visualiza un diseño moderno con paneles verticales de colores para los distintos módulos (Banco de Preguntas, Zajuna Speech, Zajuna Sign, Zajuna Chat y Text AI) acompañados de avatares 3D. En el lado derecho, está abierto el panel de herramientas para desarrolladores del navegador usando la extensión **Vue Devtools**, donde se inspecciona el árbol de componentes (mostrando etiquetas como `<RootApp>`, `<RouterView>` y `<ProjectCard>`) y las propiedades reactivas del proyecto, evidenciando que el frontend está construido con Vue.





- Interfaz Principal de Zajuna AI y Herramientas de Desarrollo de Vue.

Evidencia 2 (Navegación Lateral de Módulos): Esta captura muestra una vista diferente de la aplicación en desarrollo (localhost:7005). A la izquierda, se aprecia un menú lateral en modo oscuro que enlista los accesos a las herramientas de la plataforma mediante tarjetas, resaltando el módulo "Text AI" como el actual. El área principal de la pantalla está oscurecida. A la derecha, al igual que en la imagen anterior, se encuentra desplegado el inspector de Vue Devtools, detallando la jerarquía de componentes y los datos (props, provided, event listeners) de la vista activa, lo que demuestra la trazabilidad y la estructuración del código en Vue.

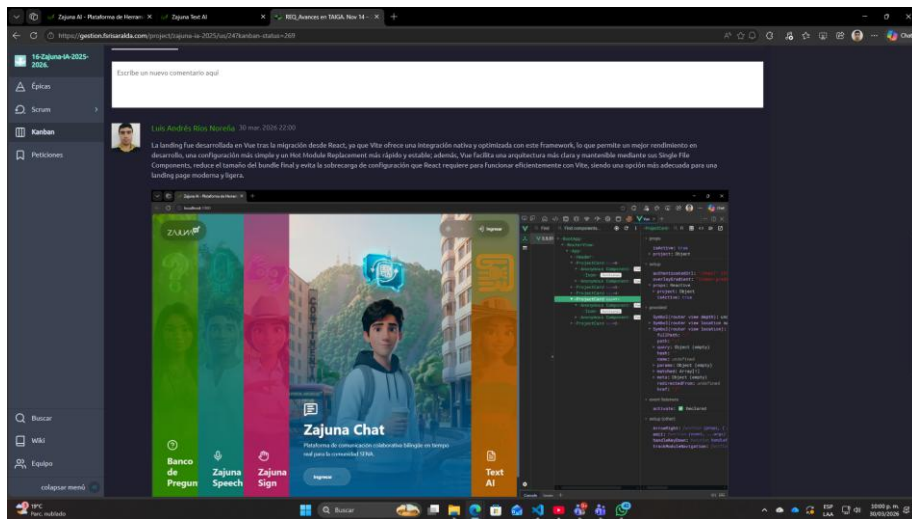


- Navegación Lateral de Módulos y Árbol de Componentes en Vue.

Evidencia 3 (Sesión presencial – despliegue de microservicios): La imagen muestra la interfaz de la herramienta de gestión de proyectos Taiga, específicamente dentro del proyecto "16-Zajuna-IA-2025-2026". Se visualiza un comentario de seguimiento realizado por ti (Luis Andrés Ríos Noreña) en el que se documenta formalmente la justificación técnica de haber migrado la landing page de React a Vue usando Vite. El texto resalta los beneficios de esta decisión (mejor rendimiento, integración nativa, Hot Module Replacement más estable y reducción del tamaño del bundle). Debajo del texto justificativo, se incluye



incrustada la captura de pantalla de la interfaz con las Vue Devtools (similar a la Evidencia 1) como prueba visual del trabajo realizado.



- Registro de Migración a Vue en la Plataforma Taiga

→ Actividad 5.

Coordinación Técnica y Sesiones de Trabajo Presencial.

Se participó en sesiones de alineación y trabajo (virtuales y presenciales) en la Fábrica de Software del SENA Risaralda para coordinar la integración de microservicios, hacer seguimiento al sprint y socializar lineamientos técnicos asociados a Zajuna AI. Durante estas sesiones se revisaron temas de trazabilidad y estado de tareas, validaciones de calidad (incluyendo QA con SonarQube), despliegue de manuales técnicos y el plan de integración UI/UX con los microservicios.

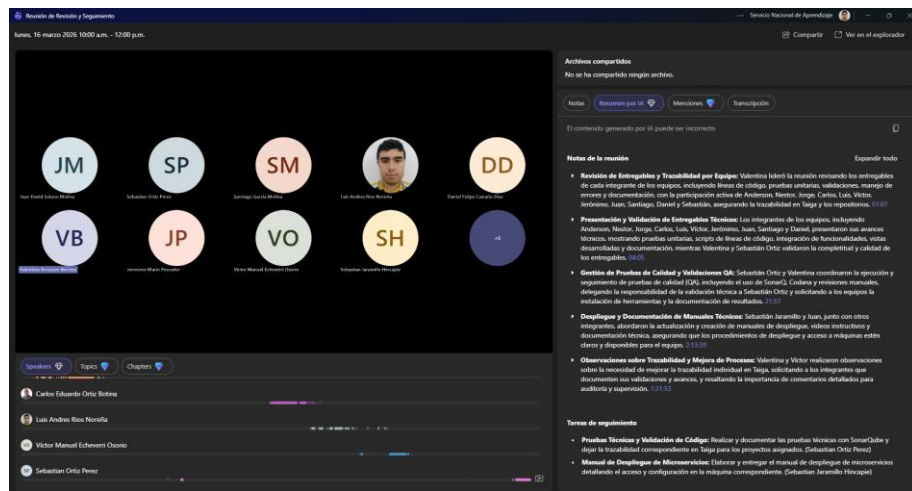
EVIDENCIAS DEL ENTREGABLE

A continuación, se puede observar las pruebas referentes a los entregables:

- **Revisión del 3er sprint.**



Evidencia 1 (Reunión de Revisión): Captura de una sesión virtual en Microsoft Teams (16 de marzo), donde se abordaron temas de trazabilidad, validaciones QA (SonarQube) y despliegue/gestión de manuales técnicos asociados al sprint.



- Reunión acerca del despliegue del proyecto microservicios para la reunión semanal

Evidencia 2 (Sesión presencial – despliegue de microservicios): Registro fotográfico de mesa de trabajo presencial en la que se discutió el despliegue del proyecto de microservicios y la coordinación para la reunión semanal.



Reunión sobre la contingencia sobre el mantenimiento del viaducto para la llegada al puesto del trabajo.

Evidencia 3 (Sesión presencial – contingencia de movilidad): Registro fotográfico asociado a la coordinación logística por la contingencia de movilidad debido al mantenimiento del viaducto, y su impacto para la llegada al puesto de trabajo.



- Reunión semanal de retroalimentación acerca del segundo sprint

Evidencia 4 (Retroalimentación): Fotografía que documenta la reunión semanal de retroalimentación del segundo sprint, donde se revisaron avances, pendientes y ajustes para la siguiente iteración.



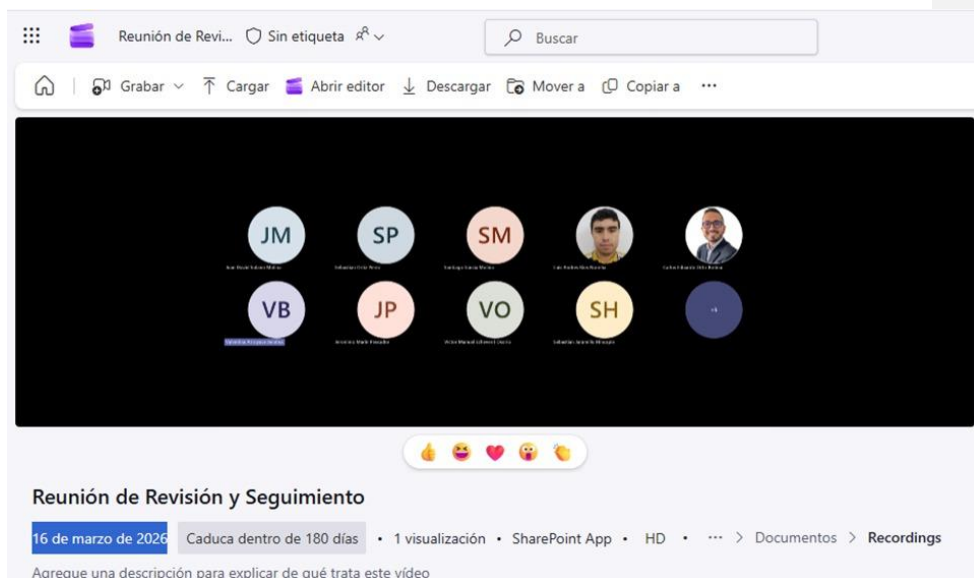


- Socializando sobre la implementación de la aplicación analítica con microservicios

Evidencia 5 (Socialización técnica): Fotografías de la socialización sobre la implementación de la aplicación analítica basada en microservicios, revisando lineamientos de integración y consideraciones técnicas con el equipo de desarrollo.



Evidencia 6 (Socialización / coordinación): Registro fotográfico de una sesión de coordinación y seguimiento con el equipo, orientada a consolidar acuerdos técnicos y próximos pasos del proyecto.



OPCIONAL FINALIZANDO EL INFORME:

NUEVAS PRACTICAS IMPLEMENTADAS (SI APLICA).

A continuación, se puede observar las pruebas referentes a la implementación de nuevas (buenas) prácticas:

MEJORAS/OBSERVACIONES (SI APLICA).

Revisado-Validado-Aprobado.