

Guía2. Desarrollo Proyecto APT Asignatura Capstone

1. Resumen avance Proyecto APT

A continuación, encontrarás distintos campos que deberás completar con la información solicitada.

Resumen de avance proyecto APT	<p><i>Durante el desarrollo del primer Sprint (23 de septiembre – 7 de octubre) se avanzó significativamente en la implementación de la infraestructura base del proyecto, cumpliendo los objetivos específicos definidos para esta etapa. Entre las principales actividades ejecutadas destacan: el levantamiento completo del ambiente en Azure, la configuración del Azure Container Registry (ACR) y Azure Web Service, la creación y conexión de la base de datos en Azure SQL, así como la elaboración del modelo entidad-relación (MER) y los scripts de base de datos para su carga inicial. Posteriormente, con el diseño ya definido de la interfaz para el frontend de la aplicación se realizó la conexión entre los endpoints del backend y la base de datos, asegurando la correcta comunicación entre los componentes del sistema. Además, se completó la documentación técnica solicitada por nuestro docente para el proyecto. En cuanto a la metodología, se mantuvo la planificación bajo el enfoque ágil SCRUM, estructurando el trabajo en Sprint semanales, lo que permitió monitorear el progreso, corregir desviaciones y optimizar el flujo de desarrollo. Los ajustes realizados fueron mínimos y se relacionaron principalmente con la integración de endpoints, extendiendo su duración. En síntesis, los objetivos específicos correspondientes a la infraestructura, base de datos, integración y documentación se encuentran cumplidos en su totalidad, lo que deja preparado el entorno para la siguiente fase de desarrollo y pruebas funcionales.</i></p>
Objetivos	<p><i>El proyecto tiene como objetivo general desarrollar e implementar una solución tecnológica basada en arquitectura cloud que integre servicios backend, base de datos y una interfaz web desplegada en Azure, orientada a optimizar la gestión y trazabilidad de la información. Para ello, se ha diseñado y levantado la infraestructura base en Azure, incorporando servicios como Azure Web Service, Azure SQL Database y Azure Container Registry. Asimismo, se modeló y desarrolló la base de datos relacional mediante la creación del modelo entidad-relación (MER) y los scripts SQL necesarios para su carga inicial. En paralelo, se implementó la</i></p>

	<p>capa backend asegurando su correcta comunicación con la base de datos a través de endpoints funcionales, y se desarrolló la interfaz frontend con un diseño responsive orientado a la experiencia del usuario. Todo el sistema fue integrado garantizando un flujo de datos estable y validado mediante pruebas en entorno de desarrollo. Además, se elaboró la documentación con los documentos entregados por nuestro docente en AVA para dejar evidencias de nuestro trabajo. El proyecto se gestionó bajo la metodología ágil SCRUM, permitiendo planificar, monitorear y controlar los avances de manera continua, asegurando la calidad técnica y el cumplimiento de los objetivos planteados.</p>
Metodología	<p>Para la ejecución y desarrollo del proyecto APT, el equipo ha adoptado la metodología Scrum con el fin de asegurar un marco de trabajo ágil y adaptativo. Esta elección estratégica permite un desarrollo iterativo e incremental, lo que garantiza que la solución final se construya sobre la base de un entendimiento profundo de los requerimientos de los usuarios y una retroalimentación constante.</p> <p>El proceso se articula en Sprint de duración definida, cada uno con objetivos claros y entregables específicos que añaden valor tangible al producto. La planificación inicial (Sprint 0) se completó con éxito, estableciendo la visión del proyecto, el levantamiento de requisitos clave de directores y apoderados, y el diseño preliminar del sistema.</p> <p>Posteriormente, la fase de desarrollo se organiza en los siguientes Sprint:</p> <p><i>Sprint 1: Foco en el módulo de gestión para directores, desarrollando la funcionalidad Core que permitirá a los colegios administrar listas y usuarios.</i></p> <p><i>Sprint 2: Implementación del sistema de notificaciones automáticas y reportes inteligentes, integrando la capa de comunicación y análisis de datos para optimizar la toma de decisiones.</i></p> <p><i>Sprint 3: Arquitectura de una base de datos segura y escalable, estableciendo la infraestructura técnica necesaria para el manejo de información sensible y el crecimiento futuro de la plataforma.</i></p> <p><i>Sprint 4: Fase de pruebas integrales, abarcando pruebas funcionales, de usabilidad y de seguridad, para garantizar la calidad, fiabilidad y robustez del software.</i></p> <p>El uso de reuniones de seguimiento, como los Daily Stand-ups y las Revisiones de Sprint, asegura una comunicación fluida, la identificación temprana de impedimentos y la validación de los avances. Este enfoque metódico no sólo maximiza la eficiencia y la productividad del equipo, sino que también permite una adaptación rápida a los cambios o nuevos requerimientos, lo que es fundamental en proyectos de desarrollo de software con un componente de innovación significativo. La culminación del proyecto incluirá una entrega final y una retrospectiva, lo que permitirá documentar las lecciones aprendidas y evaluar el proceso completo para futuras mejoras.</p>
Evidencias de avance	<p>Evidencias del avance del proyecto</p> <p>Las evidencias presentadas corresponden al Sprint 1 (23 sep – 7 oct), donde se completaron ocho actividades clave relacionadas con la infraestructura y el levantamiento del entorno en Azure y SQL. Entre ellas destacan:</p> <p>Levantamiento del ambiente en Azure (ACR, Web Service y SQL).</p> <p>Creación del modelo entidad-relación (MER) y script de base de datos.</p> <p>Configuración de repositorios de la aplicación.</p> <p>Estas evidencias demuestran el progreso técnico del proyecto APT, reflejando la correcta implementación del entorno de desarrollo y la preparación de la base de datos necesaria para las siguientes etapas.</p> <p>Resguardo de calidad</p>

La calidad del proyecto se resguarda mediante la aplicación de metodologías ágiles (SCRUM), control de versiones en repositorios y el uso de herramientas propias de la disciplina de ingeniería informática, como Azure SQL, Azure Container Registry y gestión de infraestructura en la nube.

Cada actividad fue verificada por el equipo responsable antes de marcarse como finalizada, asegurando coherencia, funcionalidad y trazabilidad del avance técnico.

Evidencia metodología Sprint 0 y 1 finalizado

SCRUM Sprint 0 2 sep – 16 sep (7 actividades)		0 0 0	Completar sprint	...
<input checked="" type="checkbox"/> SCRUM-106	Diseñar prototipo interfaz inicial	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-108	Acta de constitución del proyecto y carta Kick-off	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-111	Levantamiento de requerimientos de directores	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-112	Definir alcance del sistema	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-114	Configuración de repositorios y entorno de desarrollo (PROFESOR)	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-115	Diagrama de procesos y casos de uso	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-116	Creación de arquitectura	FINALIZADA ✓	- =	...

SCRUM Sprint 1 23 sep – 7 oct (9 actividades)		0 0 0	Completar sprint	...
<input checked="" type="checkbox"/> SCRUM-103	Levantamiento Ambiente Azure	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-104	Levantamiento Azure ACR	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-105	Levantamiento Azure Web Service	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-107	Creación Azure SQL	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-109	MER	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-110	SCRIPT bd	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-113	Creación Repositorios (APP)	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-117	Levantamiento BD en SQL	FINALIZADA ✓	- =	...
<input checked="" type="checkbox"/> SCRUM-118	Creación Plan de pruebas	TAREAS POR HACER ▾	- =	P

Evidencia levantamiento ambiente Azure

Recursos

Recente Favorito

Nombre	Tipo	Última consulta
 micolegio-rg	Grupo de recursos	hace 1 hora
 api-mi-colegio	App Service	hace 1 hora
 db-micolegio-server	SQL Server	hace 2 horas
 db-micolegio	Base de datos SQL	hace 3 horas
 micolegio	App Service	hace 1 día
 acrmicolegio	Container registry	hace 6 días

[Ver todo](#)

Evidencia levantamiento Azure ACR (Azure Container Registry)

Guía Estudiante - Desarrollo Proyecto APT

Fase 2

Inicio >

acrmicolegio Container registry

Buscar Mover Eliminar

Información general

Registro de actividad Control de acceso (IAM) Etiquetas Inicio rápido Visualizador de recursos Eventos Configuración Servicios Permisos para el repositorio Directivas Supervisión Automatización Ayuda

Información esencial

Grupo de recursos (mover) : micolegio-rg Ubicación : East US Suscripción (mover) : Suscripción de Azure 1 Id. de suscripción : 4e6bzht Azureplan Eliminación temporal (ver...) : Deshabilitado Etiquetas (editar) : Agregar etiquetas

Servidor de inicio de sesión : acrmicolegio-e4c8gjdvb4e6bzht.azurecr.io Fecha de creación : 22/9/2025, 22:33 GMT-3 Estado de aprovisionamiento : Succeeded Plan de precios : Básico Ámbito de etiqueta de recurso : Reutilización del grupo de recursos

Comenzar Supervisión Funcionalidades (9) Tutoriales

Simplifique la administración del ciclo de vida de los contenedores

Container Registry permite compilar, almacenar y administrar imágenes de contenedor y artefactos en un registro privado para todos los tipos de implementaciones de contenedor. [Más información](#)

Cloud Key Vault

Levantamiento Azure web service

Microsoft Azure

Buscar recursos, servicios y documentos (G+)

api-mi-colegio Aplicación web

Explorar Detener Intercambiar Reiniciar Eliminar Actualizar Descargar perfil de publicación Restablecer perfil de publicación

Introducción Registro de actividad Control de acceso (IAM) Etiquetas Diagnosticar y solucionar problemas Microsoft Defender for Cloud Eventos (versión preliminar) Secuencia de registro Visualizador de recursos Favoritos Cuotas Implementación Configuración Rendimiento

Essentials

Grupo de recursos (...) : micolegio-rg Estado : En ejecución Ubicación (mover) : East US 2 Suscripción (mover) : Suscripción de Azure 1 Id. de suscripción : 4e6bzht Etiquetas (editar) : Agregar etiquetas

Dominio predeterminado : api-mi-colegio-a9gjtckeebsdq.eastus2-01.azurewebsites.net Plan de App Service : ASP-micolegiorg-9adc (F1: 1) Sistema operativo : Linux Comprobación de estado : Sin configurar

Vista JSON

Propiedades Supervisión Registros Capacidades Notificaciones Recomendaciones

Aplicación web Nombre : api-mi-colegio Modelo de publicación : Contenedor Imagen de contenedor : acrmicolegio-e4c8gjdvb4e6bzht.azurecr.io/micolegio-backtest

Centro de implementación Registros de implementación Ver registros

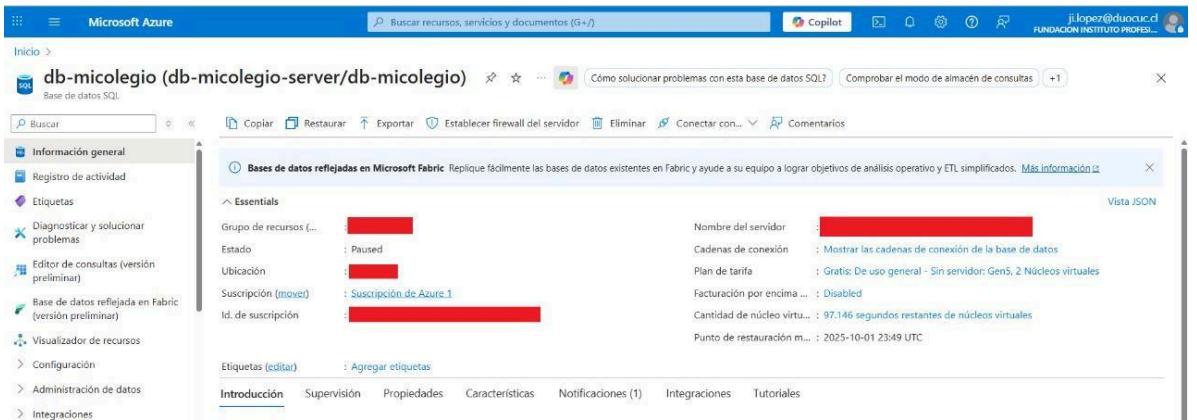
Application Insights Nombre No se admite. [Más información](#)

Dominiros Review

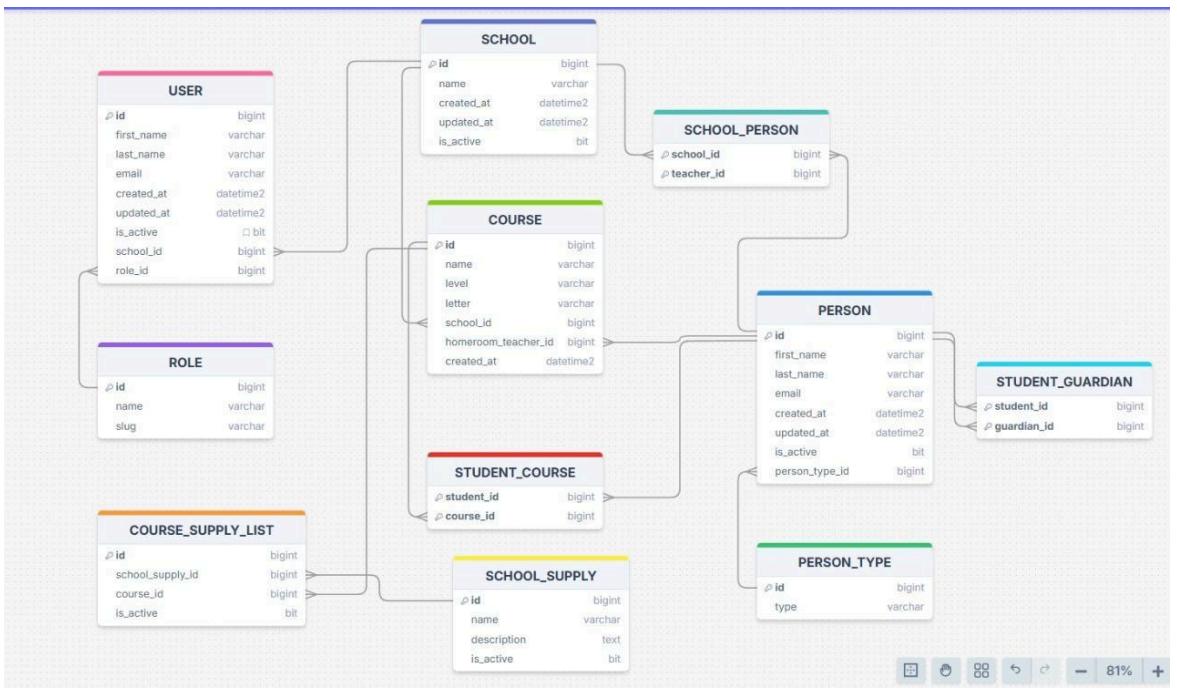
Evidencia creación Azure SQL

Guía Estudiante - Desarrollo Proyecto APT

Fase 2



Evidencia modelo entidad relación (MER)



```

    graph TD
        USER(USER) --> ROLE(ROLE)
        USER --> SCHOOL(SCHOOL)
        SCHOOL --> COURSE(COURSE)
        COURSE --> STUDENT_COURSE(STUDENT_COURSE)
        STUDENT_COURSE --> PERSON(PERSON)
        PERSON --> STUDENT_GUARDIAN(STUDENT_GUARDIAN)
        SCHOOL --> SCHOOL_PERSON(SCHOOL_PERSON)
        SCHOOL_PERSON --> PERSON
        PERSON --> PERSON_TYPE(PERSON_TYPE)
        COURSE_SUPPLY_LIST(COURSE_SUPPLY_LIST) --> SCHOOL_SUPPLY(SCHOOL_SUPPLY)
        SCHOOL_SUPPLY --> PERSON
    
```

Evidencia del script para la creación de las tablas en la BD (Nota, acá solo mostramos un fragmento del script)

Guía Estudiante - Desarrollo Proyecto APT

Fase 2

```
Script base de datos

CREATE TABLE ROLE (
    id BIGINT IDENTITY(1,1) PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    slug VARCHAR(255) NOT NULL
);

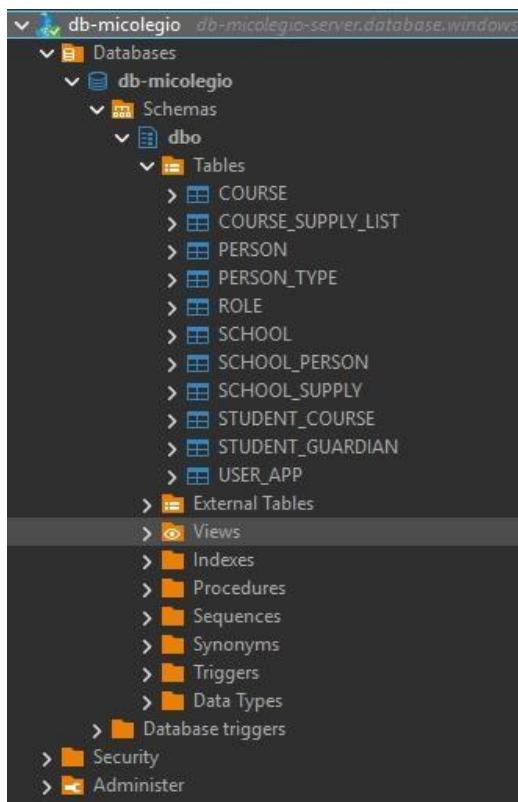
CREATE TABLE SCHOOL (
    id BIGINT IDENTITY(1,1) PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    created_at DATETIME2 NOT NULL DEFAULT GETDATE(),
    updated_at DATETIME2 NOT NULL DEFAULT GETDATE(),
    is_active BIT NOT NULL DEFAULT 1
);

CREATE TABLE USER_APP (
    id BIGINT IDENTITY(1,1) PRIMARY KEY,
    first_name VARCHAR(255) NOT NULL,
    last_name VARCHAR(255) NOT NULL,
    email VARCHAR(255) NOT NULL UNIQUE,
    created_at DATETIME2 NOT NULL DEFAULT GETDATE(),
    updated_at DATETIME2 NOT NULL DEFAULT GETDATE(),
    role_id BIGINT NOT NULL,
    CONSTRAINT FK_USER_SCHOOL FOREIGN KEY (school_id) REFERENCES SCHOOL(id),
    CONSTRAINT FK_USER_ROLE FOREIGN KEY (role_id) REFERENCES ROLE(id)
);

CREATE TABLE PERSON_TYPE (
    id BIGINT IDENTITY(1,1) PRIMARY KEY,
    type VARCHAR(255) NOT NULL
);

CREATE TABLE PERSON (
    id BIGINT IDENTITY(1,1) PRIMARY KEY,
    first_name VARCHAR(255) NOT NULL,
    last_name VARCHAR(255) NOT NULL,
    email VARCHAR(255) NOT NULL UNIQUE,
    created_at DATETIME2 NOT NULL DEFAULT GETDATE(),
    updated_at DATETIME2 NOT NULL DEFAULT GETDATE(),
    is_active BIT NOT NULL DEFAULT 1,
    person_type_id BIGINT NOT NULL,
    CONSTRAINT FK_PERSON_TYPE FOREIGN KEY (person_type_id) REFERENCES PERSON_TYPE(id)
);
```

Evidencia levantamiento de la BD en SQL



Evidencia creación de los repositorios Git Hub

The screenshot shows a GitHub search interface with the following details:

- CAPSTONE_002V_GRUPO_1** (Public)
 - Updated 1 hour ago
 - Starred
- mi-colegio.front** (Public)
 - TypeScript Updated 2 days ago
 - Star
- mi-colegio.back** (Public)
 - Java Updated last week
 - Star

Evidencia creación documento Plan de pruebas

Plan de pruebas App Mi Colegio "LOGIN"

ID DEL CASO	FUNCIONALIDAD	PRECONDICIÓN	DATO DE PRUEBA	PASO A PASO	RESULTADO ESPERADO	RESULTADO OBTENIDO	CLAVE DE PRIORIDAD
							BAJO
TC-001	Funcionalidad: Login Escenario: Si el usuario tiene correo institucional debe validar correctamente el ingreso.	Usuario creado en la plataforma. Usuario con correo institucional creado	URL:	1. Disponer de la URL creada. 2. Agregar correo institucional.	Si el usuario tiene correo valido el sistema debe permitir agregar dentro del input.		MEDIO
TC-002	Funcionalidad: Login Escenario: Si el usuario agrega correo valido y contraseña debe permitir el ingreso a la APP	Usuario creado en la plataforma. Usuario con correo institucional creado y contraseña valida	URL:	1. Disponer de la URL creada. 2. Agregar correo institucional. 3. Agregar contraseña. 4. Presionar ingresar.	Si el usuario agrega correo valido y contraseña correcta al presionar ingresar debe entrar al sistema		MEDIO
TC-003	Funcionalidad: Login Escenario: al ingresar con sus credenciales incorrectas no debe ingresar a la APP	Usuario creado en la plataforma. Usuario con correo institucional creado y contraseña valida	URL:	1. Disponer de la URL creada. 2. Agregar correo institucional. 3. Agregar contraseña invalida. 4. Presionar ingresar.	Si el usuario agrega correo invalido y contraseña correcta al presionar ingresar no debe entrar al sistema		ALTO
TC-004	Funcionalidad: Login Escenario: al ingresar con sus credenciales incorrectas no debe ingresar a la APP	Usuario creado en la plataforma. Usuario con correo institucional creado y contraseña valida	URL:	1. Disponer de la URL creada. 2. Agregar correo institucional correcto. 3. Agregar contraseña invalida. 4. Presionar ingresar.	Si el usuario agrega correo valido y contraseña incorrecta al presionar ingresar no debe entrar al sistema		ALTO
TC-005	Funcionalidad: Login Escenario: Usuario no registrado no debe ingresar al sistema	Usuario que no creado en la plataforma.	URL:	1. Disponer de la URL creada. 2. Agregar correo institucional correcto. 3. Agregar contraseña valida. 4. Presionar ingresar.	Si el usuario no existe en los registros de BD no debe ingresar al sistema		ALTO
TC-006	Funcionalidad: Login Escenario: Si el usuario agrega correo valido y contraseña debe permitir el ingreso a la APP y modificar la interfaz	Usuario creado en la plataforma. Usuario con correo institucional creado y contraseña valida	URL:	1. Disponer de la URL creada. 2. Agregar correo institucional valido. 3. Agregar contraseña valida. 4. Presionar ingresar.	Al ingresar al sistema el usuario debe ver los botones seleccionar curso crear lista		ALTO

Otras evidencias

Diseño definitivo

Despliegue de la aplicación tanto de interfaz de usuario (Front-end) como del servidor (Back-end)

Vista desde un Pc de escritorio

Aplicación MiColegio

Hogar Acerca Servicios Contacto

Aplicación Mi Colegio

Esta es la aplicación para la gestión de útiles escolares de tu colegio.

Hola desde API

Seleccionar Curso Crear Listado



Vista desde un celular

WhatsApp 07:05 49% App MiColegio

App MiColegio

Esta es la aplicación para la gestión de útiles escolares de tu colegio.

Hola desde API

Seleccionar Curso Crear Listado

.eastus2-01.azurewebsites.net

< > ⌂ ⌄ ⌅

2. Monitoreo del Plan de Trabajo

Examina cuidadosamente tu plan de trabajo, enfocándote especialmente en la columna de estado de avance y ajustes.

Plan de Trabajo

Competencia o unidades de competencias	Actividades	Recursos	Duración de la actividad	Responsable ¹	Observaciones	Estado de avance	Ajustes
Aplicar metodologías ágiles para la gestión de proyectos tecnológicos.	Levantamiento de ambiente Azure.	Cuenta Azure, documentación técnica, portal Azure.	1 semana	JL	Configuración del entorno inicial para despliegue.	Completado	Sin ajustes.
Implementar soluciones en infraestructura cloud.	Levantamiento de Azure ACR (Azure Container Registry).	Azure ACR, CLI, repositorio Docker.	2 días	JL	Se creó el contenedor base para despliegues posteriores.	Completado	Sin ajustes.
Desarrollar e integrar servicios web en entornos cloud.	Levantamiento de Azure Web Service.	Azure App Service, Docker, VS Code.	3 días	JL	Se probó despliegue inicial de servicio web	Completado	Sin ajustes.

¹ En caso de que el Proyecto APT sea grupal, en esta columna deben indicar el nombre de los responsables de cada tarea o actividad. Esto posteriormente permitirá diferenciar la evaluación por cada integrante.

Diseñar y administrar bases de datos relacionales.	Creación de Azure SQL y levantamiento de BD.	Azure SQL Database, SSMS, scripts SQL	4 días	JL	BD creada y conectada al entorno cloud.	Completado	Sin ajustes.
Modelar estructuras de datos según requerimientos funcionales.	Elaboración del MER (Modelo Entidad-Relación).	Draw SQL, documentación de requerimientos.	2 días	JL	Se definieron entidades y relaciones principales.	Completado	Sin ajustes.
Desarrollar scripts y procedimientos para bases de datos.	Creación del SCRIPT BD.	SQL Server Management Studio, archivos SQL.	2 días	JL	Scripts listos para carga inicial de datos.	Completado	Sin ajustes.
Aplicar control de versiones y buenas prácticas de desarrollo colaborativo.	Creación de repositorios (APP).	GitHub, control de versiones.	1 día	JL	Repositorio configurado para control de código y despliegues.	Completado	Sin ajustes.
Desarrollar interfaces interactivas orientadas a la experiencia de usuario.	Creación del Frontend (interfaz web).	React, HTML, CSS, JavaScript, VS Code.	1 semana	AO	Se desarrolló la interfaz principal con diseño responsive.		

3. Ajustes a partir del monitoreo

Profundiza en las observaciones de tu plan de trabajo. Analiza las actividades planificadas y señala qué aspectos facilitaron u obstaculizaron la ejecución del plan. Plantea cómo abordaste y/o abordar los obstáculos. Por último, señala los ajustes que realizaste al plan de trabajo a partir de este análisis.

Factores que han facilitado y/o dificultado el desarrollo de mi plan de trabajo:

Entre los principales factores que han facilitado el desarrollo del plan destacan la planificación por Sprint bajo metodología SCRUM, la distribución clara de tareas, y el uso de herramientas cloud integradas (Azure, GitHub, SQL Server), lo que permitió mantener un flujo de trabajo ordenado y trazable.

Además, la experiencia previa del equipo en despliegues Azure y manejo de entornos virtualizados favoreció la ejecución sin contratiempos técnicos significativos.

Como dificultades, se presentaron pequeños ajustes durante la conexión entre frontend, backend y base de datos, principalmente por diferencias en rutas de los endpoints y configuraciones de puertos. Este obstáculo fue abordado mediante revisión de logs y ajuste en los archivos de configuración del entorno.

Gracias a estas correcciones tempranas, el plan pudo mantenerse dentro de los plazos establecidos.

Actividades ajustadas o eliminadas:

No fue necesario eliminar actividades del plan original; sin embargo, se realizaron ajustes menores en la etapa de integración de endpoints, para asegurar la correcta comunicación entre los componentes.
El ajuste consistió en extender dos días la duración de esta actividad y documentar de forma más detallada los procesos de conexión, lo que permitió asegurar la calidad técnica y la trazabilidad del código.
El resto de las actividades se ejecutó conforme a la planificación inicial, gracias a la buena gestión de recursos y coordinación del equipo.

Actividades que no has iniciado o están retrasadas:

A la fecha, todas las actividades planificadas en el Sprint 1 se encuentran completadas, sin retrasos ni tareas pendientes.
Esto se logró mediante un monitoreo constante del avance semanal y el uso de herramientas colaborativas para control de versiones y comunicación técnica.
Se proyecta mantener esta dinámica para los próximos Sprint, priorizando el cumplimiento de plazos y la detección temprana de posibles desviaciones.