

EVALUACIÓN	Obligatorio 2	GRUPO	Todos	FECHA PUBLICACIÓN	15/05/2023
MATERIA	Programación 3				
CARRERA	Analista Programador – Analista en Tecnologías de la Información				
CONDICIONES	<p>- Puntos: Máximo: 35 Mínimo: 1 - Fecha máxima de entrega: 22/06/2023</p> <p>LA ENTREGA SE REALIZA EN FORMA ONLINE EN ARCHIVO NO MAYOR A 40 MB EN FORMATO ZIP, RAR O 7zip.</p> <p>IMPORTANTE:</p> <ul style="list-style-type: none">- Inscribirse- Formar grupos de hasta DOS personas.- Subir el trabajo a Gestión antes de la hora indicada, ver hoja al final del documento: "RECORDATORIO"				

INFORMACIÓN IMPORTANTE:

- Las consultas sobre la letra deberán realizarse a través del foro de aulas.
- La no asistencia a la defensa implica la pérdida de puntos del obligatorio para el estudiante. En caso de fuerza mayor deberá avisar al docente mediante mail y a coordinación por Teams para analizarse se autoriza la programación de una nueva fecha.
- Si el tamaño del archivo comprimido que entrega supera los 40Mb puede eliminar de la solución las carpetas bin, obj y packages, preservando los archivos propios que pueda haber incluido en estas carpetas (ej.: archivos de texto, imágenes, etc.). Se recomienda enfáticamente mantener una versión con los paquetes incluidos para prevenir problemas durante la descarga de paquetes.
- **El formato de la defensa se definirá previo a la entrega.**
- **Las aclaraciones realizadas por los docentes a través del foro de aulas serán consideradas parte integral de la letra, por lo que es responsabilidad de los estudiantes acceder asiduamente al foro.**

En este proyecto se deberá continuar con el sistema comenzado durante el obligatorio 1, refactorizando para utilizar arquitectura limpia.

Se mantendrá el dominio anterior, mejorándolo en caso de que lo considere necesario, y se agregará lo necesario para cumplir con los nuevos requerimientos. Incluso los datos podrán mantenerse, siempre y cuando no deba realizar cambios en el dominio que impacten sobre el esquema de las tablas que ya fueron creadas.

Se realizará usando ASP.NET Core 7, C#, Entity Framework 7 y se agregarán los clientes HTTP Client para consumir los servicios resueltos en el punto 1 utilizando Web Api .NET Core.

Las reglas de negocio referentes a las entidades anteriores se mantienen, salvo aclaración en contrario en algún requerimiento.

El desarrollo de esta etapa incluye:

- 1- Adaptar, en caso de que sea necesario, la capa de lógica del negocio para que incluya:**
 - a. Implementación de Value Object para al menos cuatro atributos
 - b. Incluir las Interfaces de los repositorios
 - c. Utilizar excepciones personalizadas para las validaciones
- 2- Agregar una capa de aplicación que incluya cada uno de los casos de uso utilizados anteriormente y los que se agregan en esta etapa.**
- 3- Adaptar la arquitectura, en caso de ser necesario, para llevarla a arquitectura limpia.**
- 4- La comunicación entre capas se realizará exclusivamente mediante abstracciones (Interfaces) utilizando Inyección de dependencias.**
- 5- Implementar las Api Web que expongan las operaciones implementadas en el obligatorio anterior.**

Se tomará en cuenta:

- El acceso a las operaciones se realizará usando el verbo HTTP apropiado según la guía REST.
- El retorno de las operaciones incluirá el código HTTP apropiado y cuando sea necesario la información en formato JSON.
- Las URL estarán orientadas a recursos y no a operaciones
- La API solamente podrá acceder a la capa de aplicación, por lo que se deberán implementar DTOs para las entidades de negocio o cualquier otro tipo de datos (no integrado con el lenguaje) que se desee enviar o recibir.
- Las APIs deberán asegurarse mediante JWT Token (ver abajo).

6- Agregar las siguientes consultas:

- a. Dados el identificador de un tipo y un monto obtener el nombre y capacidad (cantidad de huéspedes que puede alojar) de las cabañas de ese tipo que tengan un costo diario menor a ese monto, que tengan jacuzzi y estén habilitadas para reserva.
- b. Dados dos valores, obtener los mantenimientos realizados a las cabañas con una capacidad que esté comprendida (topes inclusive) entre ambos valores. El resultado se agrupará por nombre de la persona que realizó el mantenimiento, e incluirá el nombre de la persona y el monto total de los mantenimientos que realizó.

Las consultas se implementarán en Linq y su resultado será expuesto a través de una Web Api.

7- Implementar JwtToken para la seguridad de las Api Web:

Todas las apis que impliquen una operación de modificación (altas, bajas, modificaciones) deberán ser autenticadas mediante el uso de tokens JWT.

8- Documentar la/las Web Api.

Todas las API deberán estar documentadas utilizando la especificación Open Api, para lo que se sugiere utilizar una herramienta que permita realizar la documentación automática. Se deberán incluir, además de la documentación automática comentarios en los métodos que especifiquen su objetivo, parámetros, y descripción del retorno.

9- **Agregar un cliente HttpClient que consuma todas las Web Api anteriores desde una aplicación Web MVC.**

Se realizará la implementación de una nueva solución .NET con un proyecto Web ASP.NET MVC que solamente incluya lo necesario para consumir los servicios expuestos por la aplicación implementada (las funcionalidades heredadas del obligatorio anterior más las consultas agregadas en esta etapa).

Este proyecto no podrá incluir la lógica de negocio, de acceso a datos, etc., implementados. Solamente realizará y procesará las peticiones HTTP a los métodos de la Web Api.

Deberá utilizar View Model.

Deberá procesar los códigos retornados por HTTP y mostrar los mensajes de error adecuados.

ENTREGA:

DOCUMENTACIÓN:

Se incluirán en el archivo comprimido que se entrega en Gestión un único **PDF** conteniendo:

- Carátula con el nombre, número de estudiante y foto de los integrantes del grupo.
- Tabla de contenido con la página en que se ubica cada ítem de la documentación.
- Los documentos y diagramas solicitados debajo (legibles).
 - a) Diagrama de casos de uso que incluya los requerimientos anteriores y los actuales.
 - b) Diagrama de clases del dominio, incluyendo todas las entidades de negocio que modelen la realidad planteada, incluyendo el subsistema de usuarios. Se deberán indicar los atributos, relaciones, adornos relevantes de las relaciones, y **los métodos que definan su comportamiento según una correcta asignación de responsabilidades.**
Se incluirán las interfaces que utilice para las entidades de negocio y las que considere necesarias para los repositorios.
 - c) Diagrama que incluya las interfaces de los casos de uso de la capa de aplicación.
 - d) Diagrama con los DTOs utilizados
 - e) Diagrama de acceso a datos.
Incluir los repositorios concretos que considere necesarios, indicando qué interfaz implementan.
 - f) Se incluirá también el diagrama de los servicios Web Api
 - g) Diagrama con las clases del cliente MVC

- Se incluirá también un único archivo ASTAH con los diagramas.
- Scripts de la base de datos con el esquema y los datos de prueba.
- Una carpeta con imágenes para probar el alta de una cabaña durante la defensa.

IMPORTANTE: No se considerarán documentaciones que no estén entregadas en los formatos planteados (PDF y Astah), que no contengan la carátula con la identificación de los integrantes del equipo y/o que no estén incluidas dentro del único archivo comprimido entregado en gestión.

IMPLEMENTACIÓN:

- a) Solución en Visual Studio 2022 que implemente las funcionalidades mencionadas.
- b) Se deberá utilizar ASP.NET 7, MVC para la presentación, Web Api para los servicios, C# como lenguaje de programación y Entity Framework Core 7 para la persistencia en una base de datos SqlServer, usando exclusivamente Linq To Entities para las consultas de selección.

Se deberán incluir las validaciones que correspondan durante los ingresos para garantizar la consistencia del sistema, estén o no explicitadas en la letra.

Se recomienda incluir comentarios al código para comprensión de la lógica más importante.

DATOS DE PRUEBA:

Se entregarán los scripts con la creación de las tablas y los insert suficientes para probar la aplicación durante la defensa y corrección. Deberá haber al menos diez instancias de cada entidad.

Tomar en cuenta que en algunos casos – por ejemplo, en los horarios de las actividades - el límite mínimo de 10 instancias puede no ser suficiente para realizar todas las pruebas relevantes.

La no presentación de los datos de prueba, o su insuficiencia para probar la aplicación en la defensa serán valorados negativamente.

RECORDATORIO: IMPORTANTE PARA LA ENTREGA

➤ **Obligatorios** (Cap.IV.1, Doc. 220)

La entrega de los obligatorios será en formato digital online, a excepción de algunas materias que se entregarán en Bedelía y en ese caso recibirá información específica en el dictado de la misma.

Los principales aspectos a destacar sobre la **entrega online de obligatorios** son:

1. La entrega se realizará desde gestion.ort.edu.uy
2. Previo a la conformación de grupos cada estudiante deberá estar inscripto a la evaluación. **Sugerimos realizarlo con anticipación.**
3. **Cualquier integrante del grupo de obligatorio podrá subir la entrega**
4. Cada equipo (2 estudiantes) debe entregar **un único archivo en formato zip o rar** (los documentos de texto deben ser pdf, y deben ir dentro del zip o rar)
5. El archivo a subir debe tener **un tamaño máximo de 40mb**
6. Les sugerimos **realicen una 'prueba de subida' al menos un día antes**, donde conformarán el '**grupo de obligatorio**'.
7. La **hora tope para subir el archivo será las 21:00** del día fijado para la entrega.
8. La entrega se podrá realizar desde cualquier lugar (ej. hogar del estudiante, laboratorios de la Universidad, etc)
9. Aquellos de ustedes que presenten alguna dificultad con su inscripción o tengan inconvenientes técnicos, por favor contactarse con la oficina del Coordinador o por Coordinación adjunta **antes de las 20:00hs.** del día de la entrega

Si tuvieras una situación particular de fuerza mayor, debes dirigirte con suficiente antelación al plazo de entrega, al Coordinador de Cursos o Secretario Docente.