

EVALUACIÓN	Obligatorio 2	GRUPO	Todos	FECHA	19/10/2022
MATERIA	Programación 3				
CARRERA	Analista Programador – Analista en Tecnologías de la Información				
CONDICIONES	<p>- Puntos: Máximo: 35 Mínimo: 0 - Fecha máxima de entrega: 24/11/2022</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 tres 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 analizar si 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.

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 3.1.x, C#, Entity Framework 3.1.x 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 los países, selecciones, partidos, resultados e incidencias se mantienen incambiados.

Los requerimientos que nos piden implementar en esta etapa incluyen:

1. Implementación de los clientes HTTP Client que consuman la o las Apis web creadas anteriormente a través de una aplicación MVC.

- Se implementarán los clientes HTTP Client con lo necesario para consumir desde MVC .Net Core 3.1x las Apis anteriores.

Nota: El MVC deberá mostrar los mensajes adecuados según los códigos de estado HTTP obtenidos.

- **CRUD de selección**

- Se podrá dar de alta una selección a través de MVC, consumiendo las api implementadas – o mejoradas. El país de la selección deberá seleccionarse de una lista desplegable con todos los países disponibles. Esa lista se obtiene también consumiendo una API, que será necesario implementar en caso de no haberlo hecho previamente.
- Se podrá dar de baja una selección.
- Se podrán modificar los datos de la selección. Se busca la selección por su Id, se muestran sus datos y se permite modificarlos.
- Se mostrarán los datos de la selección dado su Id, incluyendo todos sus datos propios y al menos el nombre y código de su país, y su bandera.
- Se mostrarán todas las selecciones, incluyendo todos sus datos propios y al menos el nombre y código de su país y su bandera.

2. Nuevo requerimiento: Obtener la tabla de posiciones de un grupo:

- Se solicita el nombre de grupo y se despliegan en forma tabular las selecciones de ese grupo incluyendo el total de puntos obtenidos por cada una en los partidos que haya jugado (calculados según la regla de ligilla), la cantidad total de goles a favor, cantidad total de goles en contra, y diferencia de goles (a favor menos en contra).
- El listado deberá estar ordenado en forma descendente por puntaje total obtenido.
- Se deberá contar con una Api Web que retorne la tabla de posiciones solicitada, y el cliente MVC que la consuma.

3. Nuevo requerimiento: búsquedas.

- Se podrán buscar los partidos por los siguientes criterios:
 - Grupo
 - Nombre de la Selección o país de una de sus selecciones participantes
 - Código iso alfa del país de una de sus selecciones participantes
 - Entre dos fechas

Para estas búsquedas además de los métodos necesarios en la lógica de negocio/de aplicación/de acceso a datos se implementará una web api (o se incluirán controller/métodos en la anteriormente implementada) y se consumirán desde MVC a través de un cliente HTTP.

4. Implementación de un subsistema de identificación/autenticación/autorización/ por roles.

Se restringirá el acceso a las funcionalidades según un sistema de roles.

Se deberá contar con usuarios, donde cada usuario puede tener uno o más roles.

Cada usuario tendrá un Id, email (único) y password. El email será válido según las reglas habituales, la password deberá incluir al menos un carácter en minúscula, uno en mayúscula, un dígito, un carácter no alfabético de entre los siguientes: punto, coma, signo de admiración final (!). El largo será mínimo de 8 caracteres.

Los roles tendrán un Id, un nombre y una descripción. Ninguno puede ser vacío. El Id es identity. Para un mismo rol se puede contar con varios usuarios.

Si bien no será necesario implementar todas las operaciones CRUD para este sistema, sí se contará con los repositorios necesarios (se lanzará throw new NotImplementedException en caso de que la funcionalidad no sea necesaria).

Las funcionalidades que sí se deberán implementar son el Registro, Login y Logout. Se accederán a través de MVC.

Este sistema estará en un proyecto separado, y también lo estará la base de datos que albergue los usuarios.

Los valores para los roles estarán pre-cargados en la base de datos.

También se precargará un usuario admin con email : admin@qatar2022.uy y password Qatar.01

Cuando un usuario se da de alta queda automáticamente con rol "invitado". Para este prototipo se precargarán datos para al menos tres usuarios más, y al menos uno de ellos tendrá rol Apostador e invitado.

ROL	ACCESO
Admin	A todas las funcionalidades sin excepción.
Apostador	Accede a las búsquedas y a obtener un grupo con su fixture.
Invitado	Accede al fixture, a buscar una selección y al listado de todas las selecciones.
Usuario sin identificar (no tiene rol)	Solo accede al registro y a identificarse.

Una vez haya implementado el subsistema de acceso deberá verificar que los usuarios ingresen solamente a las funcionalidades que tienen habilitadas.

REQUERIMIENTOS NO FUNCIONALES:

- Se implementará usando Web Api Core 3.1, MVC Core 3.1 y C#
- Se cuidará que la arquitectura siga los lineamientos basados en DDD y Clean Architecture manejados en el curso.
- Se manejarán los códigos HTTP apropiadamente, tanto para retornar desde las web apis como para informar lo sucedido desde MVC
- Se utilizarán archivos de configuración para la configuración de la creación de la BD por parte de Entity Framework.
- Las consultas se realizarán exclusivamente utilizando LINQ (salvo en el caso de los usuarios)
- El subsistema de usuarios/roles se implementará usando ADO.NET conectado para el acceso a datos.

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
- 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 mostrar los países 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 2019 .NET Core 3.1 que implemente las funcionalidades mencionadas.**
- b) **Se deberá utilizar ASP.NET MVC 5.0 Framework para la presentación consumiendo las Apis a través de un cliente HTTP, C# y Entity Framework 3.1x para la persistencia en una base de datos SQL Server, Web Api .NET Core 3.1 para los servicios, 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 partidos- 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 (3 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.