**VALOR:** 40%

**DESCRIPCIÓN DEL PROYECTO**

* Desarrollar una solución web para un “BookStore”, la cual entre otras cosas permitirá integrar, conocer y desplegar listados de información de libros de interés. Para ello deberá recopilar una serie de libros de su elección y crear una solución compuesta por servicios web y front-end básico que permita consumir y desplegar los recursos del BookStore.
* Puede emplear sus propios recursos (hosting, base de datos, etc) y deberá provisionarla ya sea en su propio equipo o la nube (Azure, Amazon, Google).
* Asegurarse que no haya algún tipo de limitante por parte de la API para su aplicación (Clases no soportadas, licenciamiento de base de datos, trials, proxies, etc)
* De no encontrar ninguna, referirse al profesor y coordinar alguna recomendación.

**FUNCIONALIDADES DEL PROYECTO**

* Permitir el despliegue de libros e información pertinente del BookStore, incluyendo:
  + Libros (listado de los libros y gestión de los mismos)
  + Categorias (listado y registro de libros por categorias)
  + Reviews (permitir realizar un review por parte de un usuario)
  + Tendencias y novedades (basados en libros nuevos, más votados o nuevas categorias disponibles)
  + Cliente web unificado
* La solución se compondrá de 1 o más micro-proyectos abarcando las áreas mencionadas del BookStore:.
* Se deberá desplegar de manera visual información desplegada en la(s) API(s) web de manera que los recursos sean consumibles por la interfaz gráfica web para gestionar los recursos del BookStore.
* Utilizar al menos 1 mecanismo de almacenamiento persistente (memoria o preferiblemente una base de datos relacional o no-relacional - MS SQL Server, MySQL, PostgreSQL, MongoDB o algún otro a escoger).
* Habilitar un endpoint de registro para libros ya sea individual o en bloque (POST) que facilite el llenado del BookStore de forma eficaz mediante JSON con POSTMAN
* Desplegar de manera visual información desplegada en la(s) API(s) web sea consumible por la interfaz gráfica web.
* Deberá utilizar los aspectos técnicos abarcados durante el curso, incluyendo (y no limitado únicamente a):
  + Politicas CORS y aislamiento de endpoints
  + Diseño de Web APIs uniformes
  + Aplicación de principios REST, HATEOAS, entre otros
  + Definiciones Swagger a nivel de cada servicio web
  + Uso de cabeceras HTTP
  + Autenticacion y seguridad básica
  + Consumo desde clientes RESTful (Razor, HTML5, Web Apps u otras aplicaciones cliente)
  + Contenedores Docker por servicio o componente

**FECHA DE ENTREGA**

Fecha de entrega: **POR DEFINIR**

**CRITERIOS DE EVALUACIÓN**

* Estructura de la solución y proyectos
* Diseño e implementación de APIs
* Conectividad de la solución a diferentes clientes clientes potenciales (Razor, HTML5 Web Apps o algún otro mecanismo).
* Mecanismos de navegación y despliegue de información (listados, detalles, mantenimientos)
* Integración y uso de Middleware para ASP.NET Core ^2.2 (Entity Framework)
* Funcionalidades extra: Docker, Microservicios, MongoDB, MySQL, PostgreSQL.

**INDICACIONES** **DE** **ENTREGA**

Todo el código a utilizar deberá ser entregado como parte del proyecto (API, móvil, assets, etc). El proyecto deberá ser hospedado en línea mediante control de versiones en línea para su validación y comprobación. Para ello utilizar Git en alguna de sus plataformas disponibles (GitHub o bien Azure DevOps / VSTS - Visual Studio Team Services, el cual dispone de TFS y/o Git, a elegir).

Para la alternativa de versionamiento elegida deberá agregar al profesor mediante una invitación al correo utilizado durante el curso. De requerir privacidad en el repositorio indicarlo al profesor para coordinar un repositorio privado.

**OBJETIVOS DEL PROYECTO**

* TBD