

Requerimientos del Trabajo Práctico Integrador

El trabajo práctico integrador (TPI) es un trabajo que integra los contenidos dictados en la materia durante el año, el mismo debe ser realizado de manera grupal o individual, de ser grupal no puede excederse de 4 integrantes siendo 3 integrantes el número recomendado.

La evaluación y por lo tanto la calificación del TPI integrador es individual para la Aprobación Directa o la evaluación por mesa final y grupal para la aprobación no directa (regularización).

El TPI toma como punto de partida el diagrama de clases entregado por cada grupo habiendo sido corregido por los docentes de cada comisión. Los alumnos deben respetar esa estructura para construir su aplicación. Cualquier entidad y funcionalidad adyacente a la misma que no haya sido correspondientemente reflejada en el diagrama de clase no se tendrá en cuenta en la corrección.

Aprobación No Directa o Regularización

Los requerimientos del mismo están orientados a un estadio inicial del mismo en donde los componentes principales están definidos pero en donde ninguna funcionalidad precisa está completa o funcionando.

Los requerimientos formales para esta entrega son:

- Proyectos definidos según el patrón de clean architecture
- Entidades declaradas y con los principales atributos
- Repositorios definidos no haciendo falta la implementación por géneros. No hace falta tener ningún método definido
- Los servicios definidos y relacionados con los repositorios por medio de las correspondientes inyecciones de independencia. No hace falta tener ningún método definido
- Contexto definido tanto en el proyecto de infraestructura como las definiciones requeridas en el Program.cs.
- Migrations y Snapshot de EF presentes aunque sin tener que ser correspondientes a la estructura final de las entidades, con tener una única entidad reflejada está bien.
- Controladores definidos y con los servicios inyectados correspondientemente
- Authentication Controller definido aunque no requiere estar funcional
- La autenticación no requiere estar implementada más allá de la definición no funcional del Authentication controller.

Formato de entrega

La entrega se hará mediante la respuesta a una tarea publicada en el cvg cuya fecha coincide con el último día hábil de la anteúltima semana del cursado según el calendario académico. Cualquier entrega hecha luego de la fecha límite no se tendrá en cuenta.

La entrega de esta etapa es OBLIGATORIA PARA TODOS LOS ALUMNOS sin importar si son alumnos que apuntan a la Aprobación Directa o no.

La entrega consta de la publicación del link del repositorio de GitHub en donde se encuentre el código pertinente. Cualquier commit realizado a posteriori de la fecha de la entrega no se tendrá en cuenta, se corregirá el código pertinente a la fecha correspondiente. No olvidar poner el repositorio público para que pueda ser accedido por los docentes

Aprobación Directa (Promoción)

La aprobación directa de la materia implica la aprobación de la asignatura de manera formal. De esta manera la instancia de evaluación tendrá la misma rigurosidad que una evaluación en mesa por medio de un final regular.

Un TPI se encuentra en condición de aprobación directa cuando cumple los siguientes requerimientos:

- Todos y cada uno de los requerimientos de la Aprobación No Directa
- Los repositorios, servicios y controladores deben estar completos, funcionales y correctamente inyectados.
- La persistencia de datos tiene que estar operativa en una base de datos relacional mediante el paradigma code first (utilizando EF Core)
- Los repositorios deben tener implementado el patrón generic
- La API debe contar con al menos una pipeline de CI/CD funcional (puede haber más si se decide utilizar una estrategia de multiples environments) de GitHub Actions que deploye la API y los recursos subyacentes en Azure.
- El resource group de Azure dedicado a la API debe tener además de la misma una base de datos relacional multiusuario conexión (no puede ser SQL Lite) como SQL Server o MySQL
- La API debe consumir en alguno de sus endpoints expuestos al menos un servicio de terceros el cual debe estar implementado mediante HttpClient Factory
- La autenticación debe estar funcional y el secret para la generación de los token JWT debe estar alojado en al menos una variable de ambiente del Web API en Azure, siendo recomendado que el secret se encuentre alojada en un Azure Key Vault.

Formato de entrega

El trabajo debe estar desarrollado en el mismo repositorio entregado el día de la presentación para la aprobación no directa.

La evaluación del mismo se hará de manera oral e individual en las instancias de Aprobación directa (las últimas dos semanas del calendario académico) o bien en cualquier mesa de evaluación final como alumno regular.

Cuando el alumno se presente a rendir es condición excluyente que la API se encuentre deployada en una dirección de dominio accesible de manera externa. El alumno además debe presentar el código del repositorio ya sea por medio de una de las computadoras de los laboratorios como su computadora de uso personal.