Rúbrica de Evaluación de entregas del Proyecto final

**Angular**

Una rúbrica es un conjunto de criterios y normas que sirven para evaluar el nivel de desempeño en una tarea. Es una herramienta de calificación que simplifica la tarea de corrección y permite lograr una evaluación más objetiva y transparente tanto para el evaluador como el evaluado.

Adicionalmente, la rúbrica ayuda a igualar y alinear los criterios de evaluación entre los diferentes docentes y tutores.

En resumen, la rúbrica:

1. Define los criterios de evaluación de cada entrega de proyecto final.
2. Describe y detalla los puntos clave que debe tener cada entrega.
3. Facilita y transparenta la corrección de cada trabajo.

**¿Cómo se utiliza la rúbrica?🎦**[**Ver Tutorial**](https://www.loom.com/share/ad487462a0bc424cb10355989ae7abe2)

**Entrega del proyecto final**

Se debe entregar:

1. Creación de un proyecto Angular CLI con Angular.
2. Creación de componentes de layout que incluya un navbar para el menú lateral y un toolbar para el título de la aplicación
3. Creación de componentes: Lista de Alumnos y ABM de Alumnos
4. Definir formularios Reactivos de ABM de alumnos
5. Definir la lógica y estructura de representación de datos en listado, utilizando tablas de Angular Material tomando sus datos de arrays y funciones typescript

**Consigna:** Creación de un proyecto frontend basado en Angular para gestionar los asistentes a una serie de cursos

Deberá contener el perfil de administrador, el cual podrá listar, realizar altas y bajas de los alumnos, cursos e inscripciones y modificar sus datos. También podrá crear y modificar usuarios

El perfil usuario podrá listar los alumnos y los cursos, pero sí podrá agregar o eliminar inscripciones de alumnos a los cursos. No podrá realizar ninguna operación sobre los usuarios

| **Proyecto Frontend Angular (10)** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Criterios** | **Nulo**  AUSENTE  0PUNTOS | **Bajo**  CONFUSO  2 PUNTOS | **En proceso**  FALTA DE PROFUNDIDAD  3 PUNTOS | **Correcto**  PROBLEMAS PUNTUALES  4 PUNTOS | **Óptimo**  CLARO  5 PUNTOS | **NOTA** |
| Aspecto funcional | No presenta una estructura acorde y el proyecto presentado no se adecúa a lo estipulado. | Incluye la definición de las bases iniciales de un proyecto frontend basado en Angular CLI, sin embargo la segunda parte no se desarrolla. | Logra definir las bases iniciales de un proyecto frontend basado en Angular, pero integra la estructura mínima ni Angular Material un poco pobre o incompleto. | Se definen correctamente las bases iniciales de un proyecto frontend basado en Angular CLI, incluye la estructura mínima e integra Angular Material De todos modos, es preciso realizar ajustes en la aplicación y se observan oportunidades de mejora. | Logra definir las bases iniciales de un proyecto frontend basado en Angular.  El proyecto presentado con Angular CLI incluye la estructura mínima e integra Angular Material, el cual está correctamente aplicado. |  |
| Aspecto técnico | No presenta una definición y lógica en la estructura.  Los componentes y las tablas no son acordes.  No se carga en la plataforma de Coderhouse u otra plataforma. | Se presentan inconsistencias en la estructura y la lógica del proyecto.  Incluye un mínimo parcial de las vistas creadas: layout, tabla de alumnos y edición  No se utilizan correctamente las tablas de Angular Material para la inclusión de los componentes.  Las tablas dinámicas no utilizan archivos JSON o propiedades estáticas.  El formulario reactivo de ABM de alumnos no presenta interactividad ni interacción con el resto de la aplicación. La instancia de validación es incompleta o casi nula. | Se presenta en general con una estructura bien desarrollada. Se podrían realizar algunos ajustes en la lógica del proyecto.  Incluye algunos de los componentes: Layout, tabla de alumnos y edición de alumnos  Las tablas dinámicas toman los datos de propiedades estáticas dentro del código,  Hay algunas dificultades en la interacción de los formularios reactivos con el resto de la aplicación. Presentan alguna dificultad en su desarrollo y lectura. | La definición de la lógica utilizada y la estructura son correctas. Se podrían realizar mínimos ajustes para terminar de pulir.  Incluye todos los componentes: Layout, tabla de alumnos y edición de alumnos  Las tablas dinámicas toman sus datos archivos JSON  Los formularios reactivos de alumnos son correctos pero se podrían realizar algunos ajustes en las validaciones.  El proyecto fue subido gitHub | Contiene una excelente definición de la lógica utilizada y presenta una perfecta estructura.  Incluye todos los componentes: Layout, tabla de alumnos y edición de alumnos  Las tablas dinámicas toman sus datos de archivos JSON locales.  Los formularios reactivos de alumnos interactúan de forma excelente con el resto de la aplicación y están bien definidos con sus respectivas validaciones.  El proyecto fue subido a gitHub |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |