

# Proyecto Semestral

Construya un sistema de información para Transmilenio.

## Requerimientos

El sistema debe proveer las siguientes funcionalidades:

1. Conductores
  - a. CRUD conductores
  - b. Cada conductor tiene: nombre, cédula, teléfono, dirección, buses asignados
  - c. La asignación de buses debe indicar: todos los buses asignados y los días de la semana que le fueron asignados cada bus.
2. Buses
  - a. CRUD buses
  - b. Cada bus tiene: placa, modelo, rutas asignadas
    - i. La asignación de rutas debe indicar qué días de la semana está cada ruta asignada a dicho bus
  - c. Un bus no puede ser borrado si tiene conductores asignados
  - d. Un bus puede ser modificado si tiene conductores asignados
3. Rutas
  - a. CRUD rutas
  - b. Las rutas tienen los siguientes atributos: código, estaciones, horario de funcionamiento
    - i. Las estaciones se asume que están creadas previamente en la base de datos, no hace falta agregar la funcionalidad CRUD. Tampoco hay que validar que exista un camino directo entre una estación y otra.
    - ii. El horario de funcionamiento debe indicar: días de la semana, hora de inicio y fin para cada día.
  - c. Una ruta no puede ser borrada si tiene buses asignados
  - d. Una ruta puede ser actualizada aunque tenga buses asignados

El sistema tiene los siguientes tipos de usuarios con diferentes permisos:

- Administrador de rutas, el cual puede realizar la funcionalidad 3
- Coordinador, el cual puede realizar las funcionalidades 1 y 2
- Pasajero, el cual puede realizar búsquedas y consultar información sobre rutas

La interfaz de usuario del sistema debe basarse en Angular. Los datos y la lógica de la aplicación deben implementarse en un servidor Java EE o Spring. La conexión cliente-servidor debe hacerse a través de REST con JSON. Para efectos de este proyecto se debe usar la base de datos H2 que viene incluida en el servidor de aplicaciones (aunque en un proyecto real sería insuficiente).

### **1a entrega:**

- Diagrama de clases con las entidades del sistema completo

- Diagrama de navegación del sistema completo
- Aplicación multi-página (MPA) con la funcionalidad 1
- Video que explique:
  - Diseño de la aplicación
  - Funcionalidades de la aplicación y, en paralelo, qué parte del código fuente las implementan

### **2a entrega:**

- Aplicación de única página (SPA) y servicios REST que implementen todos los casos de uso, con excepción de los requerimientos de roles y seguridad
- Video que explique:
  - Funcionalidades de la aplicación y, en paralelo, qué partes del código fuente la implementan

### **3a entrega:**

- Control de acceso basado en roles. En la práctica, esto significa que debe manejarse una sesión por usuario y cada rol solo puede realizar las acciones permitidas, según el enunciado
- Pruebas automáticas a nivel del cliente web, al menos una prueba por tipo de método HTTP
- Pruebas automáticas a nivel del servidor, al menos el caso de uso más complejo
- Las pruebas automáticas deben cargar datos de prueba en la base de datos, los cuales deben removerse después de cada prueba
- Video que explique:
  - Funcionalidades de la aplicación y, en paralelo, qué partes del código fuente las implementan

### **Evaluación**

Para que el proyecto tenga una nota > 0.0 debe cumplir con lo siguiente:

- Deben subirse a Brightspace todos los entregables
- El código fuente debe haber sido almacenado en un repositorio GIT. En la práctica, el código debe entregarse en un archivo comprimido y debe incluir el directorio ``.git``.
  - El código debe haber sido actualizado (commit y push) a intervalos regulares, al menos 1 vez por semana. No se permiten actualizaciones excesivamente grandes en un solo commit.
- El código fuente debe compilar sin errores
- Una vez compilados, los programas y las pruebas deben ejecutar sin errores

- Las entregas deben ser implementadas únicamente usando lenguajes, librerías, frameworks y herramientas vistos en clases.

Si todo lo anterior se cumple, cada entrega se evaluará según las siguientes tablas. Si cualquier parte de lo entregado no cumple, la entrega será evaluada con 0.0.

La explicación en los videos entregados también forma parte de la evaluación. Debe ser clara, precisa, correcta y detallada. Por sobre todo, debe demostrar que Ud. entiende en su totalidad los componentes de la solución