

Universidad del Valle de Guatemala  
programación orientada a objetos



# Laboratorio #4

## Radios

Sebastián estrada 21405  
Gonzalo Santizo 21504

# Análisis

## Clases para modelar:

- Controlador: se utilizará para medir los inputs del usuario y comunicar esta información a las clases adecuadas.
- Vista: se utilizará como una simulación de la red social donde el usuario podrá ver el resultado de sus acciones.
- Radio: Esta clase será abstracta y contendrá los métodos generales que se utilizaran con todos los diferentes tipos de radios.
- Interfaces para carros A, C, S: Cada interfaz tendrá los métodos específicos para cada tipo de carro.
- Carros A, C, S: Estas clases concretas definirán los métodos de las interfaces y serán los objetos que serán instanciados dentro de la simulación.

## Atributos de las clases:

- En la clase radio se darán variables de tipo String que se llamen:
  - Emisora: Para guardar la emisora corriente dentro del radio.
  - EmisoraList: Un ArrayList de emisoras que el usuario ingrese.
  - Playlist: Un Arraylist de canciones que el usuario pueda "reproducir".
  - Contacts: Un Arraylist de contactos.

## Métodos de las clases:

- Para las interfaces:
  - CarA:
    - CallLastContact: Sera public y void
    - ShowBuissnesCard: Sera public y void
  - CarB:
    - Hold: Sera public y void
    - Weather: Sera public y void
  - CarS:
    - ChangeAuxSouce: Sera public y void
    - PlanTrip: Sera public y void
- Para las clases:
  - Radio que sera abstrac:
    - RaiseVolume: Como parámetro requiere un Radio, sera public y void
    - LowerVolume: Como parámetro requiere un Radio, sera public y void
    - ChangeStationUp: Como parámetro requiere un Radio, sera public y void
    - ChangeStationDown: Como parámetro requiere un Radio, sera public y void
    - SaveStation: Como parámetro requiere un ArrayList, sera public y void
    - PlayList: Como parámetro requiere un ArrayList, sera public y void
    - ChangeSongForward: Como parámetro requiere un ArrayList, sera public y void

- ChangeSongBackward: Como parámetro requiere un ArrayList, sera public y void
  - ConnectPhone: Sera public y void
  - DisconnectPhone: Sera public y void
  - ShowContacts: Sera public y void.
  - MakeACall: Sera public y void
  - toString: Sera public y String, este metodo es overrideado
  - getter y setter respectivos
- RadioA:
  - CallLastContact: Sera public y void
  - ShowBuissnesCard: Sera public y void
- RadioC:
  - Hold: Sera public y void
  - Weather: Sera public y void
- RadioS:
  - ChangeAuxSouce: Sera public y void
  - PlanTrip: Sera public y void
- Vista:
  - Vista: Método constructor
  - PrintMenuStrat: Sera public y void
  - SetRadioMod: Sera public y void
  - ModoRadio: Sera public y void
  - ModoReproduccion: Sera public y void
  - ModoTelefono: Sera public y void
  - ModoTelefonoCarrosS: Sera public y void
- Ctrl
  - Método main