# Examen II Parcial Ingeniería de Software

Desarrollar un programa que lea un csv file y lo exporte a un archivo de json o a un archivo de xml

## Estructura archivo CSV

La primera fila son los nombres de los campos. Los campos soportados son los siguientes

1. Entero (4,544,3000)
2. Texto (hola, adios123)
3. Fecha (#03/01/2017#)

Las columnas son delimitadas por commas, tomar en cuenta los caracteres de escape para las commas.

**VALIDAR QUE EL CSV ESTE BIEN FORMADO, OSEA QUE SI SON 7 columnas, todas las filas tienen 7 columnas. Si el valor no matchea a entero o fecha es string por default.**

## Estructura archivo JSON

Practicamente es un arreglo de objetos, donde los nombre de las columnas son los campos del objeto. Respetar los tipos inferidos. Osea un valor de 1 en csv es 1 en json, no “1”.

## Estructura archivo XML

El root node se llamaría Rows,que contiene de hijos nodos Row. Los row contienen los tag de los nombre de columna de CSV. Por decir si en csv nombre es carlos, se pondría <Nombre> Carlos</Nombre>, dentro del nodo Row.

## Estructura del Solution

* Migracion.Specs , specs con mocking
* Migracion.Console, app de consola que recibe un path de argumento mas si quiere json/xml y hace output a un archivo. **USAR AUTOFAC.**
* Migracion.Logic, tiene la lógica principal del todo el programa, recuerden que todas las dependencias van como abstracciones al constructor o métodos.

## Tomar en cuenta

* Usar SOLID, specialmente la S y D. La O pueden usar un patrón de diseño de fabrica para tratar los varios tipos. Por ejemplo que tal si quiero agregar flotantes? Casi seria solo agregar una clase nueva, no modificar un gran if.
* Usar un patrón de diseño de cada categoría.
* Tener varios tests, por decir
  + Validar CSV
  + Identificar los tipos en CSV
  + Generacion de JSON de cada tipo
  + Etc