# ASSIGNMENT 5: Project

This ***mandatory*** assignment is the final project of this course. The project is to be solved by a project team of 2 to 4 students. The computing engineering students (Data) can use the same groups as in the Web-technology course. The Automation- and Power students (Automasjon & Elkraft) create their own groups.

## Deadline

The teams/groups should be reported to Arne (e-mail asty@hials.no) by Friday 10th April, naming the team members, and the team leader/manager.

The project must be uploaded to Fronter by Friday 1st May 2015 no later than 23:59.

## Requirements for the submission

- Create a folder for your project.
- You can use either BlueJ or Netbeans (or both) as an IDE in this project. Make sure all files for the IDE you choose are in the project folder.
- The written report should be stored as a PDF-file in the root of the "Assignment 5"-folder. If you have screenshots/photos/images, insert these in your report instead of adding these as separate files.
- Before uploading your assignment to Fronter, make a **ZIP**-file (NOT RAR etc.) of your Assignment-folder, and name the ZIP-file on the format *Lastname_Firstname.zip* using the name of the team manager.

## Project Description:

### Introduction

The local radio station "Gloppen Nærradio" is in need of a system to keep track of all their music (on CDs, Tapes, HDs etc), jingles, commercial advertisements as well as self produced material like news and journalist reports.

These ***audio tracks*** are stored on different ***mediums*** like CDs, Vinyl LPs, tapes and as files on a computer/harddisk. The physical mediums, like the CDs, LPs and tapes, are stored in a physical library in the radio, and have an archive number making it easy to locate. This archive number is a unique number starting from 10000. To make the physical archiving easier, separate number series are being used depending on type of media. CDs are numbered from 10001 to 19999, LP records are numbered from 20001 to 29999, tapes are numbered from 30001 to 39999 and so on.

The following information is to be stored for the different types of Audio Tracks:

**Audio Tracks**

| | |
|---|---|
| Music | • Title (name of song)<br>• Duration (mm:ss)<br>• Name of Artist<br>• Date last played<br>• Number of times played<br>• Reference to the *medium* containing the music. |

| | |
|---|---|
| Advertising jingle | • Title of the jingle<br>• Duration (mm:ss)<br>• The product being advertised<br>• The company paying for the advertisement<br>• Reference to the *medium* containing the jingle. |
| News | • Title/heading of the news story<br>• News story (a short description of the news)<br>• Name of the journalist<br>• Duration (mm:ss)<br>• Date produced<br>• Date broadcasted<br>• Reference to the *medium* containing the news. |
| Sound effect | • Title/Name of soundeffect<br>• Duration (mm:ss)<br>• Description of the soundeffect<br>• Reference to the *medium* containing the sound effect. |

The following information is to be stored for the different types of Media:

**Media:**

| | |
|---|---|
| CD | • Title of the album<br>• Name of artist (if the CD is a collection of various artists, use "Various Artists" as name)<br>• Year released<br>• Record label (EMI, Polygram, Sony, Warner Music Group etc)<br>• Total duration (sum of the duration of each track)<br>• Tracks (as an ArrayList where the index in the list corresponds to the number of the track on the CD)<br>• Archive number |
| Tape | • Title of the tape<br>• Type of tape (analog/digital)<br>• Tracks (as an ArrayList where the index in the list corresponds to the number of the track on the tape)<br>• Archive number |
| File on HD | • Name of HD (ex "Radio Music Archive", "Radio Jingle Archive" etc)<br>• File name (like "Uptown Funk.mp3")<br>• Path to file (like "d:\\Music\")<br>• Filesize (in kB) |

The information above is not necessary the complete list of details needed, but should cover the most needed. Feel free to add other medias and/or types of audio-tracks too, like how would you add an audio-track which has a URL (i.e. a web-source)?

## Functional Requirements

A user should be able to do the following:

- Create a *Medium* (for instance a CD) and add the necessary information about this media.
- Add one or more tracks on the media created (like the music tracks on the CD), and add information about the tracks
- Change information already registered about a media
- Change information already registered about an audio track
- Search the archive by archive-number (note that not all medias have archive numbers…)
- Search the audio tracks by title of (or part of the title of) the audio track
- Search the audio tracks by artist (note that not all the tracks have an artist)
- List the content of the archive in different ways (all music tracks (including reference to the media), all CDs (including the tracks on the CD) etc)

# Tips and hints

In this section you will find some tips and hints to the assignment.

### *Time/Duration*

To keep track of the duration of a sound track, use the class ***java.time.Duration***. This class handles time without taking into account the date. It also provides methods to add- and subtract time.

### *Date*

To store date, use the ***java.util.GregorianCalendar***. This class handles all necessary details related to a date.

### *Collections*

It would probably be smart to have two collections; one holding all the audio tracks, and the other holding all the medias. It could also be a good idea to add a fillRegister()-method that creates a set of audio tracks and medias and adds these to the collections during startup of your application.

### *Menu system*

You should create a simple menu system, maybe in two levels; one main menu and some sub menus. For this project, use text IO (no GUI unless you feel sure…). Use the Parser-class from the "world-of-zuul" project to get ideas on how to read input from the keyboard)

- - - END - - -