Assignment **Showmaster DMX**

Date : 11dec2017

By : Dick van Kalsbeek, d.v.kalsbeek@roc-teraa.nl

Hours : 24 SBU (clock hours) each team member

Assessment : Team

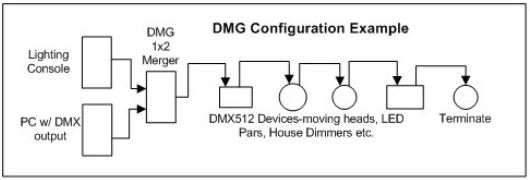
Read the assignment carefully and ask questions if anything is not clear.

# Afbeeldingsresultaat voor RGB PARIntroduction

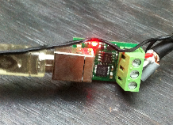
In music concerts extensive lighting is used to make the concert more spectacular. These lights are controlled by a system called DMX512. The total show is setup in a so called ‘DMX Universe’ and can contain a wide variety of devices. The system consists of a 3 or 5 wire multi-drop differential bus in which every device has an address. The system is basically quite simple where there is a maximum of 512 addresses in the bus. The ‘location’ of the data is the start address of a device, followed by the number of data packages needed for a device. Since the different devices can contain more or less functions , the address-width can differ.

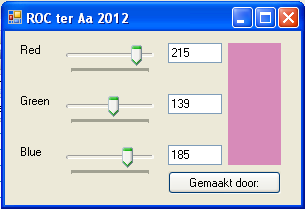
Examples:

* PAR 36 RGB spot, address-width is 6, after the start-address 5 extra addresses must be reserved
* Laser, address-width is 11, after the start-address 10 extra addresses must be reserved

Setup example:

* RIGHT: PAR spot at address 10, Laser at address 16
* RIGHT: PAR spot at address 10, Laser at address 20; NOTE: 4 addresses unused
* WRONG: PAR spot at address 10, Laser at address 12; controlling the blue for the spot affects the first address of the laser



In the previous year an application has been built named: ColorMixerDmx. The application used a DLL a hardware layer to the USB/DMX device. The DLL has been replaced by a new version and will be available during the lessons.

Customers tend to draw what they have in mind. They’re nog suppluing you with beautiful wireframes, this is what thay want you to do. Below you find a drawing of the customer who wants to have the Showmaster made by you.

**IMPORTANT**

The assignment requires Object Oriented (OO) techniques. It is probably possible to create an application like this one without using OO but this will deteriorate the maintainability and will increase the development time. OO is an absolute must in this application, creating it without OO techniques will never generate a positive grade.

# MethodWerkwijzeWerWe

This assignment is made as a team of 2 team members. A WorkbreakdownStructure(WBS) is needed to examine the required tasks, time and to locate the tasks to the developers. At the assessment the WBS is used as a guideline for more explanation of the executed tasks. So, if a developer was assigned to a task, the knowlegde of the task is expected with this developer.

The assignment is described in the projectplan. Also testplan and testresults are available at the assessment date.

# Prior knowledge & examples

During the lessons the application is explaned. Als the needed DLL and installing is guided. In the previous year a colormixer DMX was built which can be used as a reference. Also the application TuneStore, which uses object arrays has some similarity with ShowMaster.

# Requirements (MoSCoW)

In this chapter the requirements are described. This is the way which was used in the first year of the education. This is not the way customer tend to describe they’re requirements. In new assignments in the future this must be done by the software engineer.

## Must

All musts are required to receive a passing grade (a 5.5). Missing one or more musts immediately results in not passing this assignment.

## Should/Could

The shoulds and could can increase the grade when all the musts are fulfilled. Normally first the shoulds shall be completed before starting with the coulds, however this assessment does not require this way of working. All shoulds and coulds are rewarded equally.

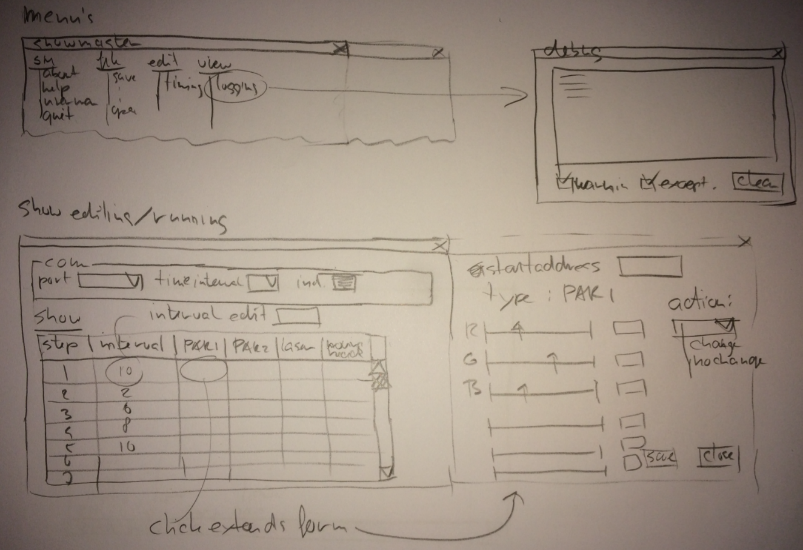
## Won’t

If a won’t is not implemented, this will result in an extra point. So, implementing a won’t will result in grade decrease.

Tips & resources

* Use classroom examples!
* To be found on N@tschool.
* Make backups!
* Websites references.
* Work together if necessary but beware! Always make sure that you understand what someone did when he helped you. A programmer should at all times be able to explain their own code.

## Example wireframes and pictures

Below some examples of a possible application.

# 

# Assessment

The assessment is based on the requirements list below. Again, in future assignments the requirements have to be defined by the software engineer.

