Documentacion proyecto 1er parcial

Table of Contents

1. Project description

In this project the user will be able to Add or Delete a user in a database sistem using tags that contains a specific color, a specific letter and a specific bar code. These tags will be readen by a camera in continuous acquisition, another module in charge of analize the data from acquisition will send it to the Database module that will Add or delete them from de sistem. The user also will be able to decide whether add or delete the user clicking in config menu and selecting the option.

2. Code component descriptions

2.1. DQMH® modules

This section describes DQMH® module responsibilities and relationships.

2.1.1. Preamble

A DQMH module is the main component of an architecture based on DQMH® framework. A DQMH module is used to implement a section of the application that has one responsibility.

DQMH® framework defines two different type of DQMH module.

Singleton:

A Singleton DQMH module can have only one instance running at any given time.

Cloneable:

A Cloneable DQMH module can have one or multiple instances running in parallel.

DQMH® framework defines two different ways to carry data throughout the application and with both other DQMH modules and non-DQMH based code.

Request events:

A request is a code that fires an event requesting the DQMH module to do something. Multiple locations in the code can send events to the DQMH module.

Request events are many-to-one.

Requests are usually named using imperative tense.

Broadcast events:

A broadcast is a code that fires an event broadcasting that the DQMH module did something. Multiple Event Structures can register to handle the Broadcast Events.

Broadcast Events are one-to-many.

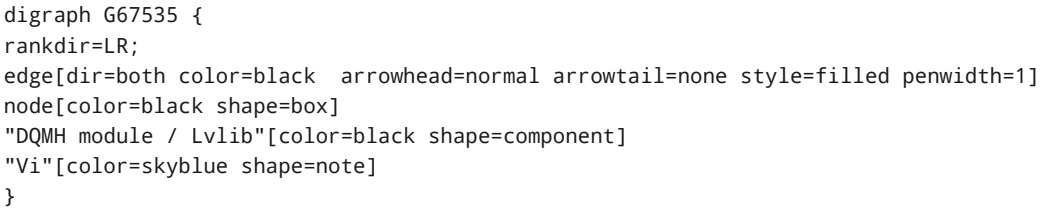
Broadcasts are usually named using past tense or passive voice.

Refer to the DQMH® framework official documentation (http://delacor.com/documentation/dqmh-html/) to find more details on how the framework works

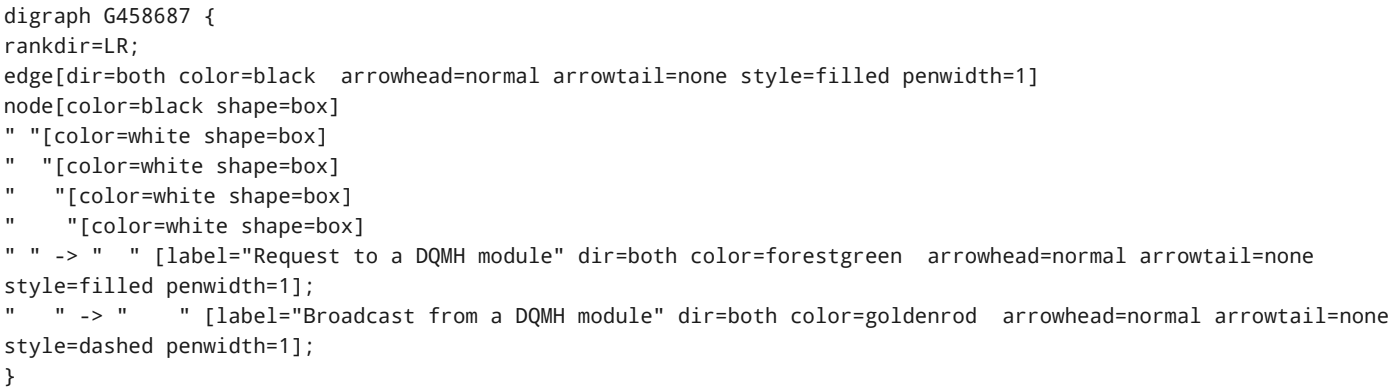
The following section gives you details on the project architecture relying on this framework. It gives you an overview of the modules' interaction and detailed information on each module.

Graphs used in this section have the following legend:

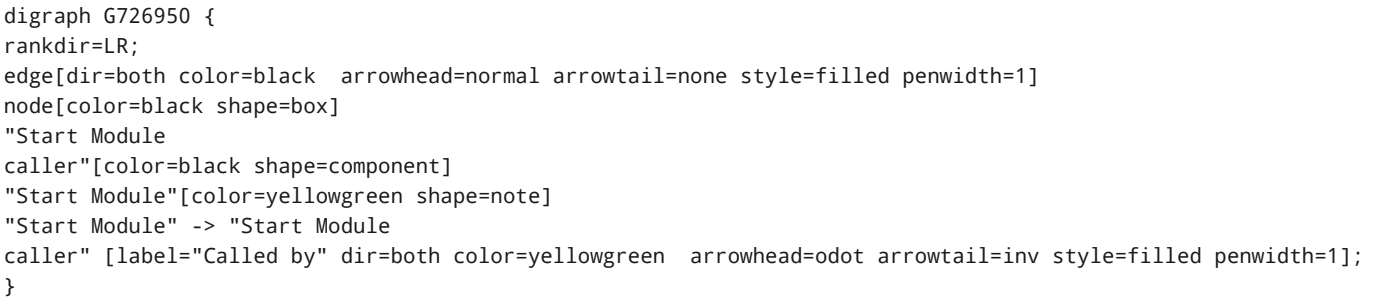
Components:



Events:



Start and Stop module callers:



```
digraph G644327 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Stop Module
caller"[color=black shape=component]
"Stop Module"[color=tomato shape=note]
"Stop Module" -> "Stop Module
caller" [label="Called by" dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
}
```

2.1.2. Modules overview

This project contains the following modules.

Table 1. Modules list

Singleton	Cloneable
Acquisition.lvlib	
Analisis.lvlib	
Configuracion.lvlib	
Database.lvlib	
Mensajes.lvlib	
UI.lvlib	

This graph represents the links between all DQMH modules.

```
digraph G19710 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"UI"[color=black shape=component]
"Acquisition"[color=black shape=component]
"Analisis"[color=black shape=component]
"Configuracion"[color=black shape=component]
"Database"[color=black shape=component]
"Mensajes"[color=black shape=component]
"Acquisition" -> "Acquisition" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"UI" -> "Acquisition" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Analisis" -> "Analisis" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Analisis" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Acquisition" -> "Analisis" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Configuracion" -> "Configuracion" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"UI" -> "Configuracion" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Database" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Analisis" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Configuracion" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Mensajes" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Configuracion" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Database" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "UI" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
}
```

2.1.3. Acquisition.lvlib

Type: Singleton

Responsibility: In this module all data will be acquire for it analisis in Analisis module

Module Start/Stop calls

```
digraph G351495 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"UI"[color=black shape=component]
"Test Acquisition API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"Acquisition"[color=black shape=component]
"Start Module" -> "UI" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Start Module" -> "Test Acquisition API" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "Acquisition" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "UI" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test Acquisition API" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted
penwidth=1];
}
```

Table 2. Start and Stop module callers

Function	Callers
Acquisition.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Acquisition API.vi
Acquisition.lvlib:Stop Module.vi	Acquisition.lvlib:Handle Exit.vi UI.lvlib:Main.vi Test Acquisition API.vi

Module relationship

```
digraph G91185 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Acquisition"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Acquisition API"[color=skyblue shape=note]
"Analisis"[color=black shape=component]
"UI" -> "Acquisition" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Acquisition API" -> "Acquisition" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Acquisition" -> "Acquisition" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Acquisition" -> "Test Acquisition API" [label=" " dir=both color=goldenrod  arrowhead=normal arrowtail=none
style=dashed penwidth=1];
"Acquisition" -> "Acquisition" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none
style=filled penwidth=1];
"Acquisition" -> "Analisis" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
}
```

Table 3. Requests callers

Request Name	Callers
Acquisition.lvlib:Show Panel.vi	Test Acquisition API.vi
Acquisition.lvlib:Hide Panel.vi	Test Acquisition API.vi
Acquisition.lvlib:Get Module Execution Status.vi	Acquisition.lvlib:Start Module.vi Acquisition.lvlib:Obtain Broadcast Events for Registration.vi
Acquisition.lvlib:Show Diagram.vi	Test Acquisition API.vi
Acquisition.lvlib:Start Acquisition.vi	UI.lvlib:Main.vi Test Acquisition API.vi

Request Name	Callers
Acquisition.lvlib:Stop Acquisition.vi	UI.lvlib:Main.vi Test Acquisition API.vi

Table 4. Broadcasts Listeners

Broadcast Name	Listeners
Acquisition.lvlib:Module Did Init.vi	Test Acquisition API.vi
Acquisition.lvlib:Status Updated.vi	Test Acquisition API.vi
Acquisition.lvlib:Error Reported.vi	Test Acquisition API.vi
Acquisition.lvlib:Module Did Stop.vi	Test Acquisition API.vi
Acquisition.lvlib:Update Module Execution Status.vi	Test Acquisition API.vi
Acquisition.lvlib:Start ACQ.vi	

Table 5. Used requests

Module	Brodcasts
Acquisition.lvlib	Acquisition.lvlib:Get Module Execution Status.vi
Analisis.lvlib	Analisis.lvlib:Start.vi

Table 6. Registred broadcast

Module	Brodcasts
—	—

2.1.4. Analisis.lvlib

Type: Singleton

Responsibility: In this VI the user will be able to see the information that the acquisition module is sending

Module Start/Stop calls

```
digraph G3762 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"UI"[color=black shape=component]
"Test Analisis API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"Analisis"[color=black shape=component]
"Start Module" -> "UI" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Start Module" -> "Test Analisis API" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "Analisis" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "UI" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test Analisis API" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted
penwidth=1];
}
```

Table 7. Start and Stop module callers

Function	Callers
Analisis.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Analisis API.vi
Analisis.lvlib:Stop Module.vi	Analisis.lvlib:Handle Exit.vi UI.lvlib:Main.vi Test Analisis API.vi

Module relationship

```
digraph G153997 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Analisis"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Analisis API"[color=skyblue shape=note]
"Acquisition"[color=black shape=component]
"Database"[color=black shape=component]
"UI" -> "Analisis" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Analisis API" -> "Analisis" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Analisis" -> "Analisis" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Acquisition" -> "Analisis" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Analisis" -> "Test Analisis API" [label=" " dir=both color=goldenrod  arrowhead=normal arrowtail=none
style=dashed penwidth=1];
"Analisis" -> "Analisis" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"Analisis" -> "Database" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
}
```

Table 8. Requests callers

Request Name	Callers
Analisis.lvlib:Show Panel.vi	Test Analisis API.vi
Analisis.lvlib:Hide Panel.vi	Test Analisis API.vi
Analisis.lvlib:Get Module Execution Status.vi	Analisis.lvlib:Start Module.vi Analisis.lvlib:Obtain Broadcast Events for Registration.vi
Analisis.lvlib:Show Diagram.vi	Test Analisis API.vi
Analisis.lvlib:Start.vi	Acquisition.lvlib:Main.vi Test Analisis API.vi
Analisis.lvlib:stop analisis.vi	Test Analisis API.vi

Table 9. Broadcasts Listeners

Broadcast Name	Listeners
Analisis.lvlib:Module Did Init.vi	Test Analisis API.vi
Analisis.lvlib:Status Updated.vi	Test Analisis API.vi
Analisis.lvlib:Error Reported.vi	Test Analisis API.vi
Analisis.lvlib:Module Did Stop.vi	Test Analisis API.vi
Analisis.lvlib:Update Module Execution Status.vi	Test Analisis API.vi

Table 10. Used requests

Module	Brodcasts
Analisis.lvlib	Analisis.lvlib:Get Module Execution Status.vi
Database.lvlib	Database.lvlib:Database info.vi

Table 11. Registred broadcast

Module	Brodcasts
—	—

2.1.5. Configuracion.lvlib

Type: Singleton

Responsibility: In this VI the user would be able to select if they want to add or delete a user saved in the database.

Module Start/Stop calls

```
digraph G204910 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"UI"[color=black shape=component]
"Test Configuracion API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"Configuracion"[color=black shape=component]
"Start Module" -> "UI" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Start Module" -> "Test Configuracion API" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "Configuracion" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test Configuracion API" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted
penwidth=1];
}
```

Table 12. Start and Stop module callers

Function	Callers
Configuracion.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Configuracion API.vi
Configuracion.lvlib:Stop Module.vi	Configuracion.lvlib:Handle Exit.vi Test Configuracion API.vi

Module relationship

```
digraph G325891 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Configuracion"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Configuracion API"[color=skyblue shape=note]
"Database"[color=black shape=component]
"Mensajes"[color=black shape=component]
"UI" -> "Configuracion" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Configuracion API" -> "Configuracion" [dir=both color=forestgreen  arrowhead=normal arrowtail=none
style=filled penwidth=1];
"Configuracion" -> "Configuracion" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Configuracion" -> "Test Configuracion API" [label=" " dir=both color=goldenrod  arrowhead=normal arrowtail=none
style=dashed penwidth=1];
"Configuracion" -> "Configuracion" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none
style=filled penwidth=1];
"Configuracion" -> "Database" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none
style=filled penwidth=1];
"Configuracion" -> "Mensajes" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none
style=filled penwidth=1];
}
```

Table 13. Requests callers

Request Name	Callers
Configuracion.lvlib:Show Panel.vi	UI.lvlib:Main.vi Test Configuracion API.vi
Configuracion.lvlib:Hide Panel.vi	Test Configuracion API.vi
Configuracion.lvlib:Get Module Execution Status.vi	Configuracion.lvlib:Start Module.vi Configuracion.lvlib:Obtain Broadcast Events for Registration.vi
Configuracion.lvlib:Show Diagram.vi	Test Configuracion API.vi

Table 14. Broadcasts Listeners

Broadcast Name	Listeners
Configuracion.lvlib:Module Did Init.vi	Test Configuracion API.vi
Configuracion.lvlib:Status Updated.vi	Test Configuracion API.vi
Configuracion.lvlib:Error Reported.vi	Test Configuracion API.vi

Broadcast Name	Listeners
Configuracion.lvlib:Module Did Stop.vi	Test Configuracion API.vi
Configuracion.lvlib:Update Module Execution Status.vi	Test Configuracion API.vi

Table 15. Used requests

Module	Brodcasts
Configuracion.lvlib	Configuracion.lvlib:Get Module Execution Status.vi
Database.lvlib	Database.lvlib:config database.vi
Mensajes.lvlib	Mensajes.lvlib:Recibir mensajes.vi

Table 16. Registred broadcast

Module	Brodcasts
—	—

2.1.6. Database.lvlib

Type: Singleton

Responsibility: This module has all the instructions and operation to save the information in the Database

Module Start/Stop calls

```
digraph G259565 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"UI"[color=black shape=component]
"Test Database API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"Database"[color=black shape=component]
"Start Module" -> "UI" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Start Module" -> "Test Database API" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "Database" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "UI" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test Database API" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted
penwidth=1];
}
```

Table 17. Start and Stop module callers

Function	Callers
Database.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Database API.vi
Database.lvlib:Stop Module.vi	Database.lvlib:Handle Exit.vi UI.lvlib:Main.vi Test Database API.vi

Module relationship

```
digraph G386660 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Database"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Database API"[color=skyblue shape=note]
"Analisis"[color=black shape=component]
"Configuracion"[color=black shape=component]
"Mensajes"[color=black shape=component]
"UI" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Database API" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Database" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Analisis" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Configuracion" -> "Database" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Database" -> "Test Database API" [label=" " dir=both color=goldenrod  arrowhead=normal arrowtail=none
style=dashed penwidth=1];
"Database" -> "Database" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"Database" -> "Mensajes" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
}
```

Table 18. Requests callers

Request Name	Callers
Database.lvlib:Show Panel.vi	Test Database API.vi
Database.lvlib:Hide Panel.vi	Test Database API.vi
Database.lvlib:Get Module Execution Status.vi	Database.lvlib:Start Module.vi Database.lvlib:Obtain Broadcast Events for Registration.vi

Request Name	Callers
Database.lvlib:Show Diagram.vi	Test Database API.vi
Database.lvlib:Database info.vi	Analisis.lvlib:Main.vi Test Database API.vi
Database.lvlib:config database.vi	Configuracion.lvlib:Main.vi Test Database API.vi
Database.lvlib:Save.vi	UI.lvlib:Main.vi Test Database API.vi

Table 19. Broadcasts Listeners

Broadcast Name	Listeners
Database.lvlib:Module Did Init.vi	Test Database API.vi
Database.lvlib:Status Updated.vi	Test Database API.vi
Database.lvlib:Error Reported.vi	Test Database API.vi
Database.lvlib:Module Did Stop.vi	Test Database API.vi
Database.lvlib:Update Module Execution Status.vi	Test Database API.vi

Table 20. Used requests

Module	Brodcasts
Database.lvlib	Database.lvlib:Get Module Execution Status.vi
Mensajes.lvlib	Mensajes.lvlib:Recibir mensajes.vi

Table 21. Registred broadcast

Module	Brodcasts
—	—

2.1.7. Mensajes.lvlib

Type: Singleton

Responsibility: In this module all mesages the program has will be shown

Module Start/Stop calls

```
digraph G824523 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"UI"[color=black shape=component]
"Test Mensajes API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"Mensajes"[color=black shape=component]
"Start Module" -> "UI" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Start Module" -> "Test Mensajes API" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "Mensajes" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "UI" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test Mensajes API" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted
penwidth=1];
}
```

Table 22. Start and Stop module callers

Function	Callers
Mensajes.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Mensajes API.vi
Mensajes.lvlib:Stop Module.vi	Mensajes.lvlib:Handle Exit.vi UI.lvlib:Main.vi Test Mensajes API.vi

Module relationship

```
digraph G1001338 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Mensajes"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Mensajes API"[color=skyblue shape=note]
"Configuracion"[color=black shape=component]
"Database"[color=black shape=component]
"UI" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Mensajes API" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Mensajes" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Configuracion" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Database" -> "Mensajes" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"Mensajes" -> "Test Mensajes API" [label=" " dir=both color=goldenrod  arrowhead=normal arrowtail=none
style=dashed penwidth=1];
"Mensajes" -> "Mensajes" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
}
```

Table 23. Requests callers

Request Name	Callers
Mensajes.lvlib:Show Panel.vi	Test Mensajes API.vi
Mensajes.lvlib:Hide Panel.vi	Test Mensajes API.vi
Mensajes.lvlib:Get Module Execution Status.vi	Mensajes.lvlib:Start Module.vi Mensajes.lvlib:Obtain Broadcast Events for Registration.vi
Mensajes.lvlib:Show Diagram.vi	Test Mensajes API.vi
Mensajes.lvlib:Recibir mensajes.vi	Configuracion.lvlib:Main.vi Database.lvlib:Main.vi UI.lvlib:Main.vi Test Mensajes API.vi

Table 24. Broadcasts Listeners

Broadcast Name	Listeners
Mensajes.lvlib:Module Did Init.vi	Test Mensajes API.vi
Mensajes.lvlib:Status Updated.vi	Test Mensajes API.vi

Broadcast Name	Listeners
Mensajes.lvlib:Error Reported.vi	Test Mensajes API.vi
Mensajes.lvlib:Module Did Stop.vi	Test Mensajes API.vi
Mensajes.lvlib:Update Module Execution Status.vi	Test Mensajes API.vi

Table 25. Used requests

Module	Brodcasts
Mensajes.lvlib	Mensajes.lvlib:Get Module Execution Status.vi

Table 26. Registred broadcast

Module	Brodcasts
—	—

2.1.8. UI.lvlib

Type: Singleton

Responsibility: In this module the user would be able to see Acquisition, Analisis and Messages Main Vi's.

Module Start/Stop calls

```
digraph G8576 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"Test UI API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"UI"[color=black shape=component]
"Start Module" -> "Test UI API" [dir=both color=yellowgreen  arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "UI" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test UI API" [dir=both color=tomato  arrowhead=odot arrowtail=inv style=dotted penwidth=1];
}
```

Table 27. Start and Stop module callers

Function	Callers
UI.lvlib:Start Module.vi	Test UI API.vi

Function	Callers
UI.lvlib:Stop Module.vi	UI.lvlib:Handle Exit.vi Test UI API.vi

Module relationship

```
digraph G218833 {
rankdir=LR;
edge[dir=both color=black  arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"UI"[color=slateblue shape=component]
"Test UI API"[color=skyblue shape=note]
"Acquisition"[color=black shape=component]
"Configuracion"[color=black shape=component]
"Database"[color=black shape=component]
"Mensajes"[color=black shape=component]
"Test UI API" -> "UI" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "UI" [dir=both color=forestgreen  arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Test UI API" [label=" " dir=both color=goldenrod  arrowhead=normal arrowtail=none style=dashed
penwidth=1];
"UI" -> "Acquisition" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"UI" -> "Configuracion" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"UI" -> "Database" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"UI" -> "Mensajes" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"UI" -> "UI" [label="   " dir=both color=forestgreen  arrowhead=onormal arrowtail=none style=filled penwidth=1];
}
```

Table 28. Requests callers

Request Name	Callers
UI.lvlib:Show Panel.vi	Test UI API.vi
UI.lvlib:Hide Panel.vi	Test UI API.vi
UI.lvlib:Get Module Execution Status.vi	UI.lvlib:Start Module.vi UI.lvlib:Obtain Broadcast Events for Registration.vi
UI.lvlib:Show Diagram.vi	Test UI API.vi

Table 29. Broadcasts Listeners

Broadcast Name	Listeners

Broadcast Name	Listeners
UI.lvlib:Module Did Init.vi	Test UI API.vi
UI.lvlib:Status Updated.vi	Test UI API.vi
UI.lvlib:Error Reported.vi	Test UI API.vi
UI.lvlib:Module Did Stop.vi	Test UI API.vi
UI.lvlib:Update Module Execution Status.vi	Test UI API.vi

Table 30. Used requests

Module	Brodcasts
Acquisition.lvlib	Acquisition.lvlib:Start Acquisition.vi Acquisition.lvlib:Stop Acquisition.vi
Configuracion.lvlib	Configuracion.lvlib:Show Panel.vi
Database.lvlib	Database.lvlib:Save.vi
Mensajes.lvlib	Mensajes.lvlib:Recibir mensajes.vi
UI.lvlib	UI.lvlib:Get Module Execution Status.vi

Table 31. Registred broadcast

Module	Brodcasts
—	—

2.2. Libraries

This section describes the libraries contained in the project.

2.3. Classes

This section describes the classes contained in the project.

3. VI descriptions

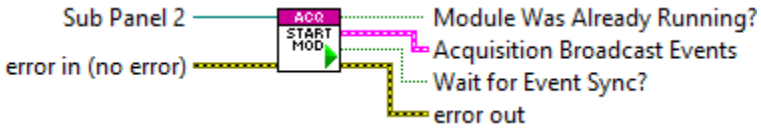
3.1. DQMH® modules

This section describes DQMH® modules events.

3.1.1. Acquisition.lvlib

Acquisition.lvlib:Start Module.vi

Event type: Not a DQMH Event

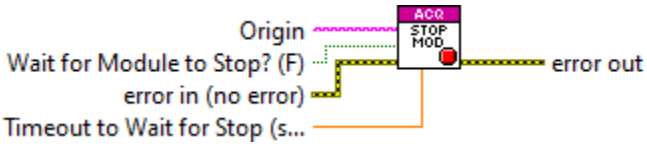


Description:

Launches the Module Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Stop Module.vi

Event type: Not a DQMH Event



Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Show Panel.vi

Event type: Request

Description:

Send the Show Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Hide Panel.vi

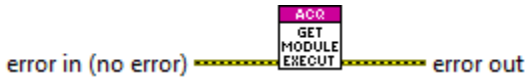
Event type: Request

Description:

Send the Hide Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Get Module Execution Status.vi

Event type: Request



Description:

Fire the Get Module Execution Status request. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Show Diagram.vi

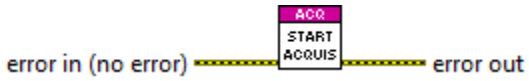
Event type: Request

Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Start Acquisition.vi

Event type: Request

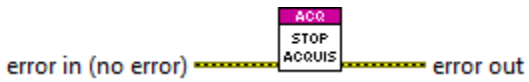


Description:

Start the camera acquisition to grab images. _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Acquisition.lvlib:Stop Acquisition.vi

Event type: Request

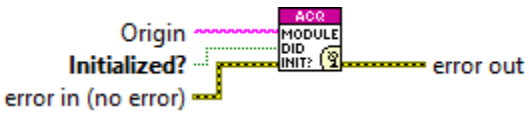


Description:

Stops the acquisition in course _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Acquisition.lvlib:Module Did Init.vi

Event type: Broadcast

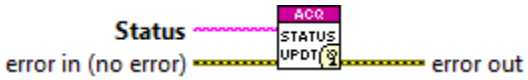


Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Status Updated.vi

Event type: Broadcast

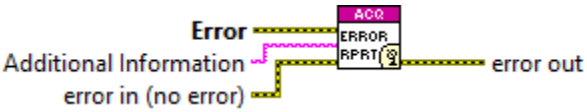


Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Error Reported.vi

Event type: Broadcast



Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Module Did Stop.vi

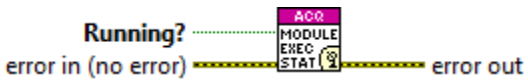
Event type: Broadcast

Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Update Module Execution Status.vi

Event type: Broadcast



Description:

Broadcast event to specify whether or not the module is running. _____ Based on Delacor QMH Project Template 5.0.0.82.

Acquisition.lvlib:Start ACQ.vi

Event type: Broadcast

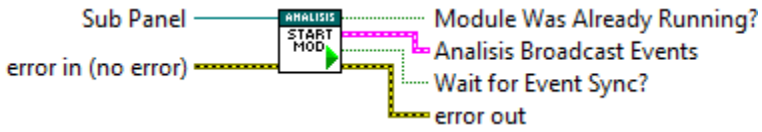
Description:

Starts acquisition broadcast _____ Created using Delacor QMH Event Scripter 5.0.0.112.

3.1.2. Analisis.lvlib

Analisis.lvlib:Start Module.vi

Event type: Not a DQMH Event

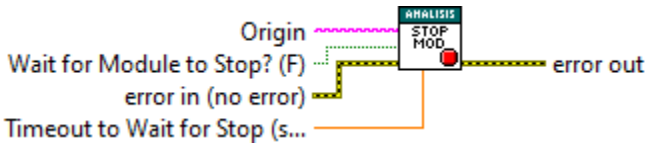


Description:

Launches the Module Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Stop Module.vi

Event type: Not a DQMH Event



Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Show Panel.vi

Event type: Request

Description:

Send the Show Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Hide Panel.vi

Event type: Request

Description:

Send the Hide Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Get Module Execution Status.vi

Event type: Request

Description:

Fire the Get Module Execution Status request. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Show Diagram.vi

Event type: Request

Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Start.vi

Event type: Request

Description:

Start module to analize samples _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Analisis.lvlib:stop analisis.vi

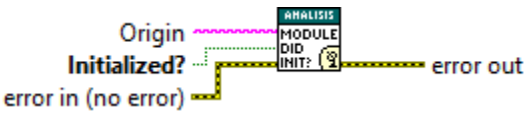
Event type: Request

Description:

Stop analisis _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Analisis.lvlib:Module Did Init.vi

Event type: Broadcast



Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Status Updated.vi

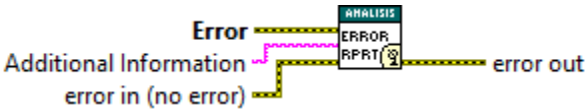
Event type: Broadcast

Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Error Reported.vi

Event type: Broadcast



Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Module Did Stop.vi

Event type: Broadcast

Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Analisis.lvlib:Update Module Execution Status.vi

Event type: Broadcast

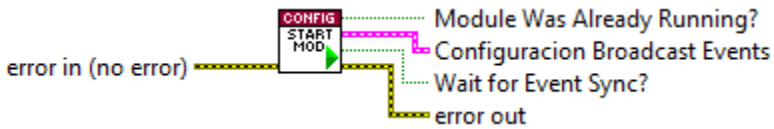
Description:

Broadcast event to specify whether or not the module is running. _____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.3. Configuracion.lvlib

Configuracion.lvlib:Start Module.vi

Event type: Not a DQMH Event

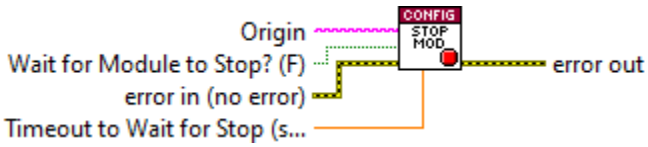


Description:

Launches the Module Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Stop Module.vi

Event type: Not a DQMH Event



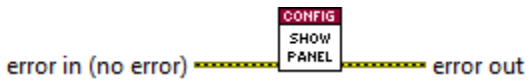
Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout**

to Wait for Stop value is ignored if 'Wait for Module to Stop?' is set to FALSE. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Show Panel.vi

Event type: Request



Description:

Send the Show Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Hide Panel.vi

Event type: Request

Description:

Send the Hide Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Get Module Execution Status.vi

Event type: Request

Description:

Fire the Get Module Execution Status request. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Show Diagram.vi

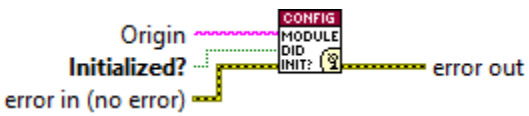
Event type: Request

Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Module Did Init.vi

Event type: Broadcast

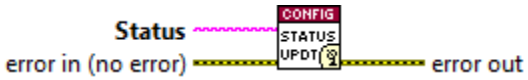


Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Status Updated.vi

Event type: Broadcast

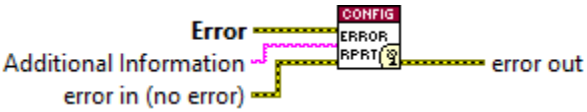


Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Error Reported.vi

Event type: Broadcast



Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Module Did Stop.vi

Event type: Broadcast

Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Configuracion.lvlib:Update Module Execution Status.vi

Event type: Broadcast

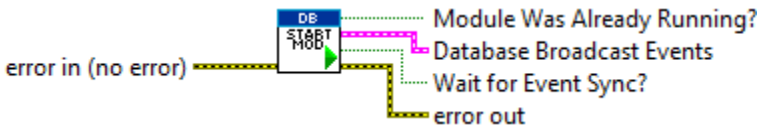
Description:

Broadcast event to specify whether or not the module is running. _____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.4. Database.lvlib

Database.lvlib:Start Module.vi

Event type: Not a DQMH Event

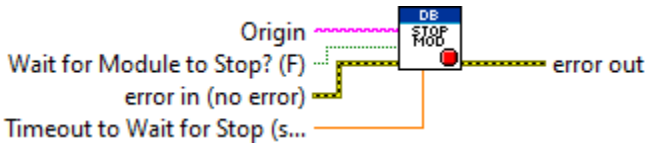


Description:

Launches the Module Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Stop Module.vi

Event type: Not a DQMH Event



Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Show Panel.vi

Event type: Request

Description:

Send the Show Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Hide Panel.vi

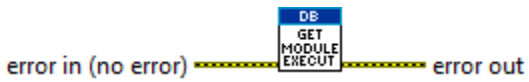
Event type: Request

Description:

Send the Hide Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Get Module Execution Status.vi

Event type: Request



Description:

Fire the Get Module Execution Status request. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Show Diagram.vi

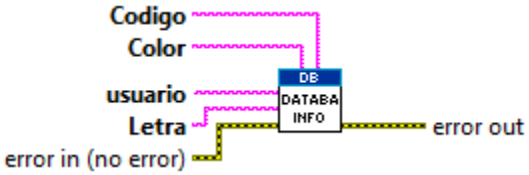
Event type: Request

Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Database info.vi

Event type: Request

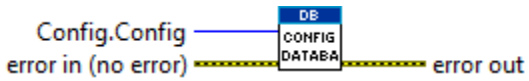


Description:

This event grabs the information of analisis and saves it in the Database _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Database.lvlib:config database.vi

Event type: Request



Description:

This event indicates whether you want to save or delete an user _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Database.lvlib:Save.vi

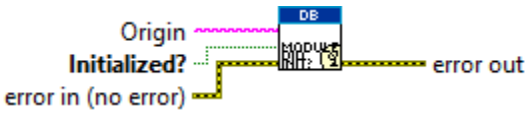
Event type: Request

Description:

Allows the UI to save the information in database _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Database.lvlib:Module Did Init.vi

Event type: Broadcast



Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Status Updated.vi

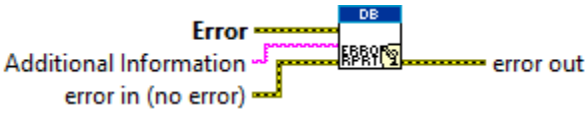
Event type: Broadcast

Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Error Reported.vi

Event type: Broadcast



Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Module Did Stop.vi

Event type: Broadcast

Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Database.lvlib:Update Module Execution Status.vi

Event type: Broadcast

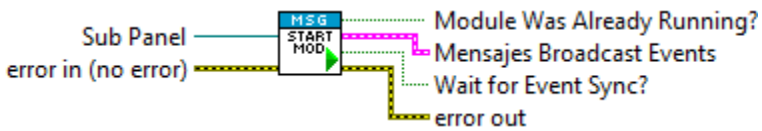
Description:

Broadcast event to specify whether or not the module is running. _____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.5. Mensajes.lvlib

Mensajes.lvlib:Start Module.vi

Event type: Not a DQMH Event

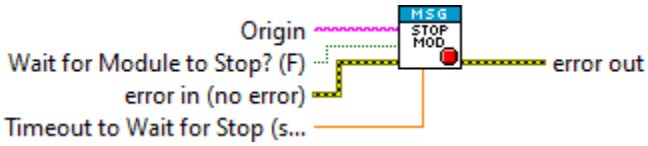


Description:

Launches the Module Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Stop Module.vi

Event type: Not a DQMH Event

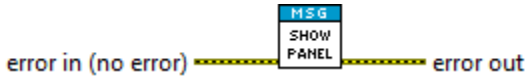


Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Show Panel.vi

Event type: Request



Description:

Send the Show Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Hide Panel.vi

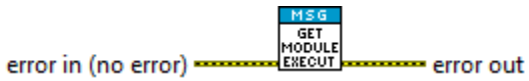
Event type: Request

Description:

Send the Hide Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Get Module Execution Status.vi

Event type: Request

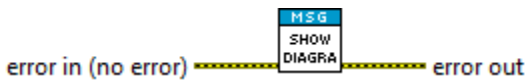


Description:

Fire the Get Module Execution Status request. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Show Diagram.vi

Event type: Request

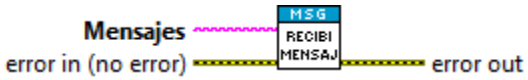


Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Recibir mensajes.vi

Event type: Request

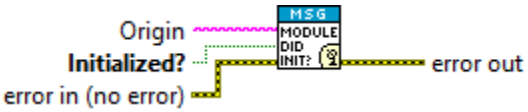


Description:

This event print messages from other modules in the string indicator of messages module _____ Created using Delacor QMH Event Scripter 5.0.0.112.

Mensajes.lvlib:Module Did Init.vi

Event type: Broadcast

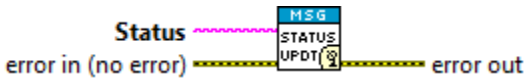


Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Status Updated.vi

Event type: Broadcast

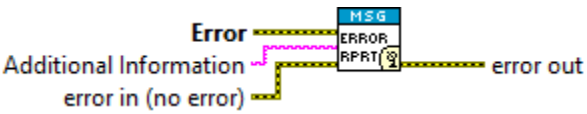


Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Error Reported.vi

Event type: Broadcast

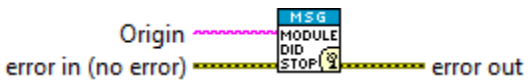


Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Module Did Stop.vi

Event type: Broadcast



Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.lvlib:Update Module Execution Status.vi

Event type: Broadcast

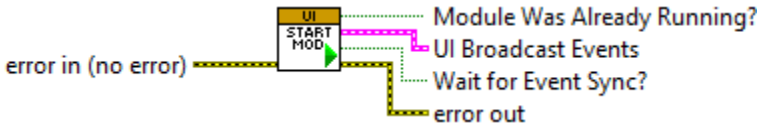
Description:

Broadcast event to specify whether or not the module is running. _____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.6. UI.lvlib

UI.lvlib:Start Module.vi

Event type: Not a DQMH Event

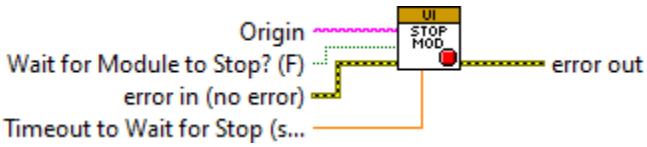


Description:

Launches the Module Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Stop Module.vi

Event type: Not a DQMH Event



Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Show Panel.vi

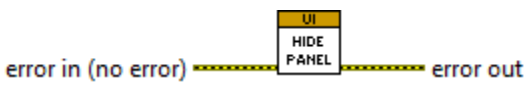
Event type: Request

Description:

Send the Show Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Hide Panel.vi

Event type: Request

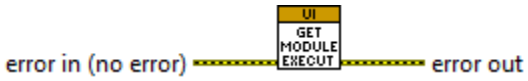


Description:

Send the Hide Panel request to the Module's Main.vi. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Get Module Execution Status.vi

Event type: Request



Description:

Fire the Get Module Execution Status request. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Show Diagram.vi

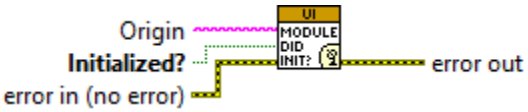
Event type: Request

Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Module Did Init.vi

Event type: Broadcast



Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Status Updated.vi

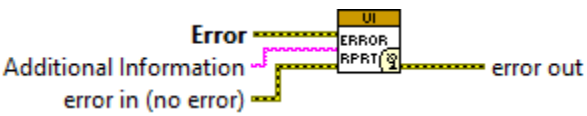
Event type: Broadcast

Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Error Reported.vi

Event type: Broadcast



Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Module Did Stop.vi

Event type: Broadcast

Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. _____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Update Module Execution Status.vi

Event type: Broadcast

Description:

Broadcast event to specify whether or not the module is running. _____ Based on Delacor QMH Project Template 5.0.0.82.

3.2. Libraries

This section describes libraries public VIs.

3.3. Classes

This section describes classes public VIs.

4. Legal Information

4.1. Document creation

This document has been generated using the following tools.

4.1.1. Antidoc

Project website: Antidoc (https://wovalab.gitlab.io/open-source/labview-doc-generator/)

Maintainer website: Wovalab (https://wovalab.com)

BSD 3-Clause License

Copyright © 2019, Wovalab, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.1.2. Asciidoc for LabVIEW™

Project website: Asciidoc toolkit (https://wovalab.gitlab.io/open-source/asciidoc-toolkit/)

Maintainer website: Wovalab (https://wovalab.com)

BSD 3-Clause License

Copyright © 2019, Wovalab, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.1.3. Graph Builder

Project website: Graph Builder (https://gitlab.com/cgambini/graph-builder)

BSD 3-Clause License

Copyright (c) 2020, Cyril GAMBINI All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.2. Product used in the project

The documented project has been developed with the following products.

4.2.1. DQMH®

Copyright © 2015-2020 by Delacor, LLC. All Rights Reserved.

Find more details on Delacor (https://delacor.com/products/dqmh/) website