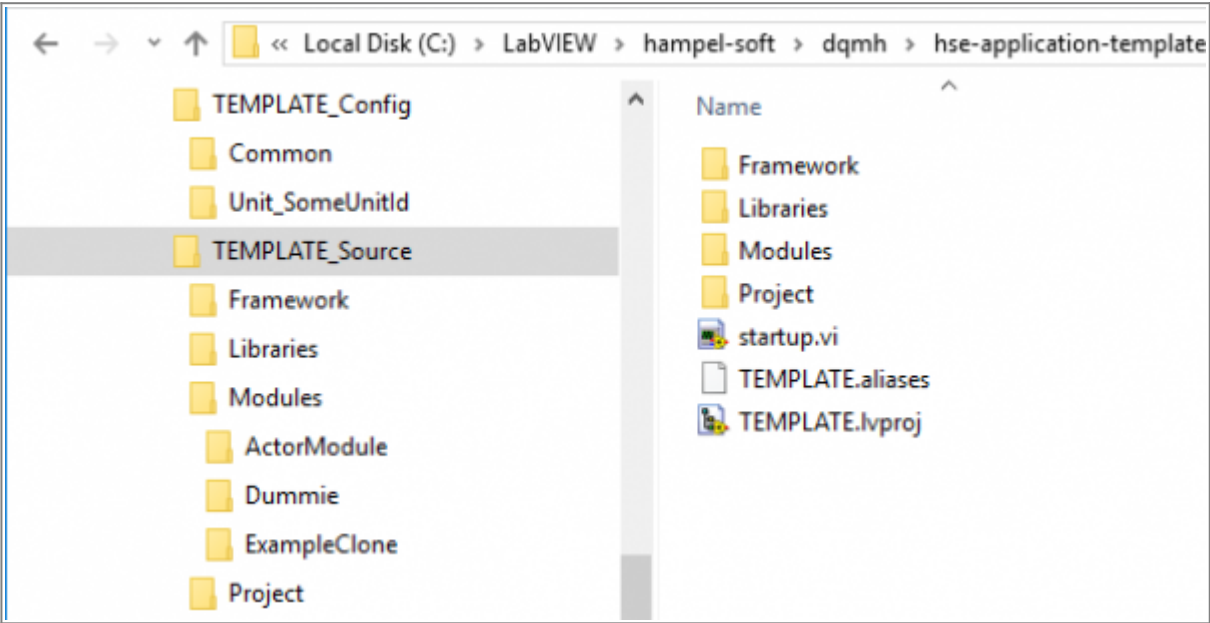


# 11 Project Structure

## Introduction

This page defines the default project structure used by most of our projects. We have also information on our [default repo structure](#) and our [overall software structure](#). Screenshots are included from an example repository for demonstrative purposes.



Throughout this page, we will use TEMPLATE as a placeholder for the actual project's name.

## Basic Structure

|                   |                                  |
|-------------------|----------------------------------|
| /TEMPLATE/        | built LabVIEW application (.exe) |
| /TEMPLATE_Config/ | all configuration data           |
| /TEMPLATE_Data/   | output data (eg measurement)     |
| /TEMPLATE_Source/ | LabVIEW source code (files)      |

## 'Config' directory

|                   |                               |
|-------------------|-------------------------------|
| /TEMPLATE_Config/ |                               |
| /Common           | common config files           |
| /Unit_xyz         | one set of configuration data |

|                                 |                                                     |
|---------------------------------|-----------------------------------------------------|
| <code>/Unit_abc</code>          | another set of configuration data                   |
| <code>config.ini</code>         | defines which unit directory to use (not under SCC) |
| <code>config.ini.default</code> | default file (checked into SCC)                     |

Contains all configuration data that's needed by the application.

---

## 'Data' directory

```
/TEMPLATE_Data/  
template-application.log
```

Contains the application's log file. Serves as a container for all sorts of files created by the application, like measurements, reports, etc.

---

## 'Source' directory

```
/TEMPLATE_Source/  
  /Framework/  
  /Libraries/  
  /Modules/  
  /Project/  
  TEMPLATE.lvproj  
  TEMPLATE_Main.vi      Main VI for Real-Time applications  
  startup.vi            Main VI for Windows applications
```

Contains the following directories and files:

## 'Framework' directory

```
/TEMPLATE_Source/  
  /Framework/  
    /Event Manager/  
    /Sidebar/  
    /UI Manager/
```

For Windows Applications: Contains generic, as-is reusable modules. Currently, we use the following modules:

- Event Manager
- Sidebar
- UI Manager



More information on framework modules: [Application Template](#)

## 'Libraries' directory

```
/TEMPLATE_Source/  
  /Libraries/  
    /hse-core-libs/  
    /hse-layout-manager/  
    /hse-logger/  
    /hse-networking/
```

Contains libraries shared by projects.



More information on our libraries: [HSE-Libraries](#)

## 'Modules' directory

```
/TEMPLATE_Source/  
  /Modules/  
    /ActorModule/  
    /Dummie/  
    /Dummie2/  
    /User Interface/
```

Contains DQMH modules utilised by the project.



More information on DQMH can be found in our knowledge base:  
[Delacor Queued Message Handler](#)

## 'Project' directory

```
/TEMPLATE_Source/  
  /Project/  
    /Project.lvlib  
    /PROJECT_ApplicationVersion--constant.vi  
    /PROJECT_InitLogging.vi  
    /PROJECT_Layout-VI.vi  
    /PROJECT_Name--constant.vi
```

```
/PROJECT_RunTimeMenu.rtm  
/PROJECT_SplashScreen.vi  
/PROJECT_StartupSteps.ctl  
/PROJECT_UserCredentials.vi
```

Contains all project-specific files that are not part of a DQMH module or a library. The content of this directory varies from project to project.

## 'TEMPLATE.lvproj'

LabVIEW project file, containing the VIs in XML format.

## Main VI

Either `startup.vi` (Windows applications) or `TEMPLATE_Main.vi` (Real-Time applications).

## Actual Source Code



The source code for the project shown can be found at  
<https://gitlab.com/hampel-soft/dqmh/hse-application-template>

From:

<https://dokuwiki.hampel-soft.com/> - **HAMPEL SOFTWARE ENGINEERING**

Permanent link:

<https://dokuwiki.hampel-soft.com/code/common/project-structure?rev=1589541123>

Last update: **2020/05/15 11:12**

