

```
#include "kontrolinterface.h"

kontrolinterface::kontrolinterface()
{
    //Opsætter server og styringsmodul
    m_pServer = new DataServer;
    m_pStyringsmodul = new styringsmodul;

    //Opsætter MW
    m_pMainWindow = new MainWindow;
    //m_pMainWindow->setWindowFlags(Qt::WindowStaysOnTopHint);
    //m_pMainWindow->setWindowFlags(Qt::AA_DontUseNativeMenuBar);
    //m_pMainWindow->showFullScreen();
    m_pMainWindow->show();

    //Forbindelser imellem slots og signals
    QObject::connect(m_pMainWindow, SIGNAL(manuChanged(Level)),
m_pStyringsmodul, SLOT(manualLeveling(Level)));
    QObject::connect(m_pMainWindow, SIGNAL(activateAutoClicked()),
m_pStyringsmodul, SLOT(activateAutomaticRegulation()));
    QObject::connect(m_pStyringsmodul, SIGNAL(SMConnectionChangedTo(bool)),
m_pMainWindow, SLOT(showSMConnectionChange(bool)));
    QObject::connect(m_pServer, SIGNAL(serverUpdateStatus(bool)), m_pMainWindow,
SLOT(showServerUpdateStatus(bool)));
    QObject::connect(m_pStyringsmodul,
SIGNAL(ManuelLevelingAck(bool)),m_pMainWindow, SLOT(displayManuText(bool)));
    QObject::connect(m_pMainWindow, SIGNAL(programTerminated()), this,
SLOT(destroyMe()));
    QObject::connect(this, SIGNAL(updateSignal(status)), m_pServer,
SLOT(updateServerSlot(status)));
    QObject::connect(this, SIGNAL(updateSignal(status)), m_pMainWindow,
SLOT(updateGuiSlot(status)));

    //Opsætter timer for server
    serverTimer = new QTimer(this);
    connect(serverTimer, SIGNAL(timeout()), this, SLOT(update()));
    serverTimer->start(UPDATE_TIME); //Starter timer
}

kontrolinterface::~kontrolinterface()
{
    delete m_pStyringsmodul;
    delete m_pMainWindow;
    delete m_pServer;
    delete serverTimer;
    delete guiTimer;
}

void kontrolinterface::destroyMe()
{
    if (m_pStyringsmodul->programTermination())
        delete this;
    else
        m_pMainWindow->displayCloseFailure();
}

void kontrolinterface::update()
{
    recentValues = m_pStyringsmodul->getStatus();
    recentValues.SecondsSinceLastGuiUpdate = m_pMainWindow-
>getSecondsSinceLastGuiUpdate();
    emit this->updateSignal(recentValues);
}
```