Filename: gpumonitor.cpp

Header:

#include <QMainWindow>

#include <QSettings>

#include <QSystemTrayIcon>

#include <QThread>

#include <QtCharts/QChartView>

#include <QtCharts/QLineSeries>

#include <QtCharts/QCategoryAxis>

#include <QDateTimeAxis>

#include <QTimer>

#include <QDebug>

#include <QMessageBox>

#include <QMenu>

#include <QMenuBar>

#include <QCloseEvent>

#include <QLibrary>

#include <QDir>

#include <QFileDialog>

#include <QDateTimeAxis>

#include <QBarCategoryAxis>

#include <QScrollBar>

#include <QDateTime>

#include "nvidiaapi.h"

#include "structures.h"

Class name: GPUMonitorThrd

Friend class: GeneralTest

Inherits: QThread

Description: GPUMonitorThrd is an abstract class that calls API provided by GPU drivers to fetch real-time GPU status such as temperature, core clock, VRAM clock, fan speed, and power.

Constructor:

**GPUMonitorThrd**(QObject\* p = Q\_NULLPTR):

Parameters: Pointer to the parent object.

Return: None.

Task: Constructs a GPUMonitorThrd object.

Key Protected Fields:

|  |  |
| --- | --- |
| float refresh\_rate = 3 | The refresh rate (seconds) of GPU information. |
| QDateTime last\_refresh = QDateTime() | The last time of GPU information refresh. |

Public Methods:

virtual QList<GPUInfo> ***getStatus***() = 0:

Parameters: None.

Return: A list of GPU information.

Task: Call API provided by GPU driver to fetch and pack GPU information.

Signals:

void **gpusInfoSignalRefresh**(QList<GPUInfo> gpusinfo):

Task: Broadcast a list of newly fetched GPU information.

void **gpuInfoSignal**(unsigned int gpucount

, unsigned int maxgputemp

, unsigned int mingputemp

, unsigned int maxfanspeed

, unsigned int minfanspeed

, unsigned int maxmemclock

, unsigned int minmemclock

, unsigned int maxgpuclock

, unsigned int mingpuclock

, unsigned int maxpowerdraw

, unsigned int minpowerdraw

, unsigned int totalpowerdraw);

Task: Broadcast a set of newly fetched overall GPU information.

Class name: NvMonitorThrd

Friend class: GeneralTest

Inherits: GPUMonitorThrd

Description: NvMonitorThrd is the implementation of the abstract class GPUMonitorThrd for NVIDIA GPU. It uses two APIs from NVIDIA. One is NVML for GPU monitoring, the other is NVIDIA API for overclocking.

Constructor:

**NvMonitorThrd**(QObject\* p = Q\_NULLPTR, NvidiaAPI \*nvapi = NULL):

Parameters: Pointer to the parent object; Pointer to the NVIDIA API.

Return: None.

Task: Constructs a NvMonitorThrd object.

Key Protected Fields:

|  |  |
| --- | --- |
| NvidiaNVML\* \_nvml | The NVIDIA NVML API for GPU monitoring. |
| NvidiaAPI \*\_nvapi | The NVIDIA API for GPU overclocking. |

Public Methods:

void ***run***() override:

Parameters: None.

Return: None.

Task: Periodically fetch GPU information by calling getStatus() method and emits signals for the MainWindow.

QList<GPUInfo> ***getStatus***() override:

Parameters: None.

Return: A list of newly fetched GPU information.

Task: Fetch and pack GPU information by calling NVML API.