TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**Project 3**

**TÌM HIỂU REPORTBUILDER VÀ ỨNG DỤNG VÀO CHƯƠNG TRÌNH   
THEO DÕI Ổ CỨNG BẰNG S.M.A.R.T**

Sinh viên thực hiện: **Vũ Tuấn Anh**

Lớp CN-CNTT K57

Giáo viên hướng dẫn: Th.S. **Nguyễn Đức Tiến**

HÀ NỘI 06 -2014

# PHIẾU GIAO NHIỆM VỤ MÔN HỌC

**1. Thông tin về sinh viên**

Họ và tên sinh viên: Vũ Tuấn Anh

Điện thoại liên lạc: 01676614597 Email:tuananh.vga@gmail.com

Lớp: CN-CNTT K57 Hệ đào tạo: Đại học chính quy

**2. Sản phẩm mục tiêu của môn học**

Tìm hiểu về công cụ ReportBuilder của Microsoft SQLServer, áp dụng để phát triển chương trình lấy thông tin về trạng thái của ổ đĩa cứng bằng công nghệ S.M.A.R.T.

**3. Các tính năng**

**Các tính năng bắt buộc:**

* Lập trình/Tìm hiểu để hoàn thành sản phẩm: Có
* Sử dụng một kho repository để lưu trữ các phiên bản của project: Có

<https://github.com/tuananhvga/SMART-Monitor>

**Các tính năng tùy chọn:**

|  |  |
| --- | --- |
| Tính năng | Có/Không |
| Sử dụng các đối tượng trong toolbox như button, text box, combo box, list box | Có |
| Lấy thông tin về trạng thái của ổ cứng | Có |
| Dự đoán khả năng ổ cứng có thể gặp vấn đề | Có |
| Hiển thị thông tin trên GUI | Có |
| Đặt lịch trình lấy dữ liệu theo thời gian | Có |
| Cảnh báo khi ổ cứng gặp vấn đề | Không |
| Tạo báo cáo về tình trạng ổ cứng | Có |
| Kết nối database SQLServer/MySQL/PostgreSQL/MongoDB | Có |
| Giao diện Ribbon | Không |

Tính năng khác nếu có: Thông tin về ổ cứng theo thời gian được lưu trữ trong cơ sở dữ liệu sử dụng MS SQLServer. Công cụ ReportBuilder được sử dụng để tạo báo cáo về trạng thái ổ cứng, ngoài ra, chương trình còn có dạng service chạy ngầm để lấy thông tin theo thời gian và bộ cài đặt đi kèm.

[PHIẾU GIAO NHIỆM VỤ MÔN HỌC 2](#_Toc418931415)

[DANH MỤC HÌNH VẼ 4](#_Toc418931416)

[DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ 4](#_Toc418931417)

[PHẦN I: NỀN TẢNG VÀ CÔNG NGHỆ PHÁT TRIỂN ỨNG DỤNG 5](#_Toc418931418)

[1.1. Các sản phẩm tương tự 5](#_Toc418931419)

[1.2. Các công nghệ kỹ thuật liên quan 6](#_Toc418931420)

[1.2.1. Microsoft Visual Studio 6](#_Toc418931421)

[1.2.2. GitHub 6](#_Toc418931422)

[1.2.3. WMI (Windows Management Instrumentation) 6](#_Toc418931423)

[1.2.4. S.M.A.R.T (Self-Monitoring, Analysis and Reporting Technology) 6](#_Toc418931424)

[1.2.5. ReportBuilder 7](#_Toc418931425)

[PHẦN 2:PHÂN TÍCH THIẾT KẾ 8](#_Toc418931426)

[2.1. Mô hình kết nối 8](#_Toc418931427)

[2.2. Cơ sở dữ liệu 8](#_Toc418931428)

[2.3. Mô hình giao diện 9](#_Toc418931429)

[2.4. Các thành phần chức năng 9](#_Toc418931430)

[PHẦN 3:SẢN PHẨM KẾT QUẢ 11](#_Toc418931431)

[3.1. Lưu ý 11](#_Toc418931432)

**DANH MỤC HÌNH VẼ**

[Hình 1 Mô hình ứng dụng 8](#_Toc418931440)

[Hình 2 Cấu trúc cơ sở dữ liệu 9](#_Toc418931441)

[Hình 3 Giao diện chính 9](#_Toc418931442)

[Hình 4 Giao diện chương trình 11](#_Toc418931443)

[Hình 5 Giao diện mục report 11](#_Toc418931444)

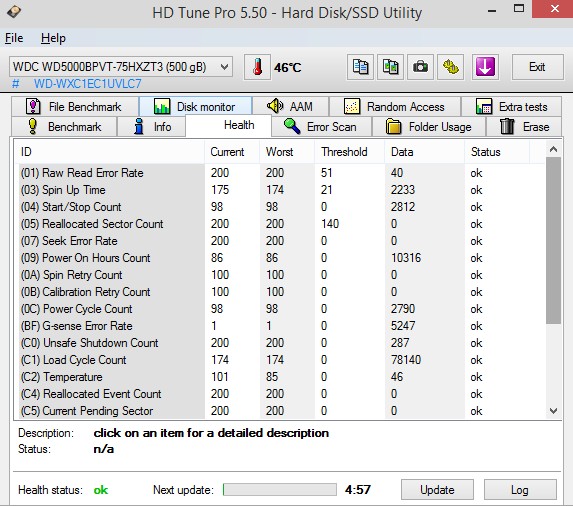
**DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ**

|  |  |  |
| --- | --- | --- |
| **Số thứ tự** | **Từ viết tắt** | **Ý nghĩa** |
| 1 | WMI | Windows Management Instrumentation |
| 2 | SMART | Self-Monitoring, Analysis and Reporting Technology |
| 3 | HDD | Hard Disk Drive |

# PHẦN I: NỀN TẢNG VÀ CÔNG NGHỆ PHÁT TRIỂN ỨNG DỤNG

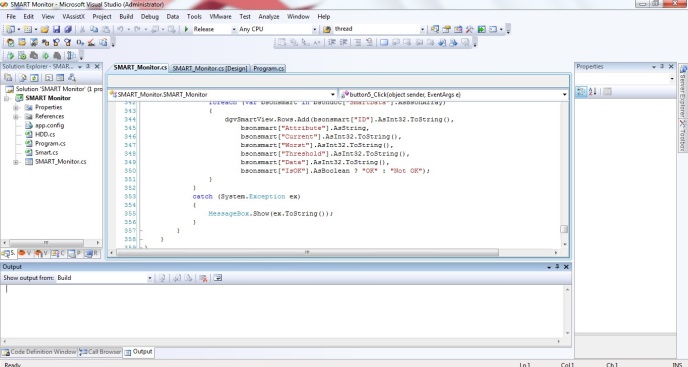
## Các sản phẩm tương tự

HD Tune

**

## Các công nghệ kỹ thuật liên quan

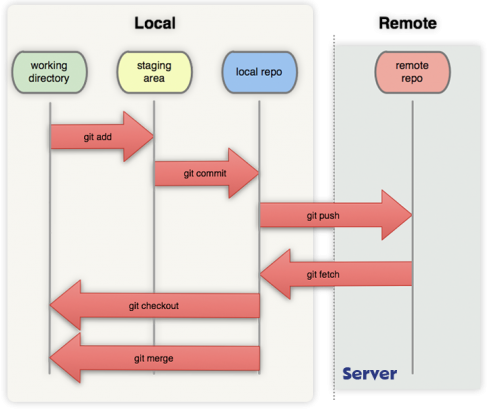
### Microsoft Visual Studio

Là một công cụ đa năng, đặc biệt là trong việc tạo lập và phát triển phần mềm.

MVS hỗ trợ rất nhiều ngôn ngữ lập trình (C++,C#,VB…), công cụ (Spy++, Error lookup…), khả năng mở rộng cao với các extension và còn là một công cụ hỗ trợ debug rất mạnh

Trong đồ án này, Microsoft Visual Studio được sử dụng để tạo giao diện và phần thực thi của chương trình.

### GitHub

Github [http://github.com](http://github.com/), còn được gọi là social network dành cho developer đi vào hoạt động tháng 2 năm 2008, là một dịch vụ sử dụng hệ thống quản lý phân tán GIT giúp người dùng lưu trữ source code cho các dự án.

GitHub cung cấp dịch vụ thương mại và cả tài khoản miễn phí cho các dự án nguồn mở. Theo khảo sát của người sử dụng Git vào năm 2009, Github hiện đang là server Git lưu trữ source code phổ biến nhất hiện nay

Trong đồ án này, GitHub được dùng để lưu trữ mã nguồn và báo cáo môn học.

### WMI (Windows Management Instrumentation)

Windows Management Instrumentation là phần mở rộng của Windows Driver Model cung cấp một lớp giao tiếp giữa các thành phần hạ tầng với hệ điều hành. Nó cho phép các script có thể lấy thông tin, hoặc điều khiển các tác vụ hệ thống,

Trong đồ án này, WMI là công nghẹ được sử dụng để tương tác với tính năng S.M.A.R.T của ổ cứng.

### S.M.A.R.T (Self-Monitoring, Analysis and Reporting Technology)

SMART là công nghệ tự động giám sát tình trạng hoạt động của ổ đĩa cứng để thông qua BIOS, các phần mềm có thể thông báo cho người dùng biết trước các hư hỏng có thể xảy ra để xử lý (sao lưu dữ liệu, thay ổ cứng…).

### ReportBuilder

ReportBuilder là công cụ hỗ trợ cho việc tạo các báo cáo, sử dụng với hệ quản trị CSDL MS SQLServer. Các báo cáo có thể được thiết kế vô cùng đơn giản bởi ReportBuilder hỗ trợ kéo-thả các đối tượng, hành động, thiết lập điều kiên.. rất nhiều.

Nên sử dụng ReportBuilder phiên bản 3.1 và SQLServer Express R2 with Advanced Tools để có thể kết nối được với nhau. (ReportBuilder 3.1 không kết nối được với bản SQLServer nào thấp hơn).

Trong đồ án này, ReportBuilder được sử dụng để tạo ra báo cáo về tình trạng ổ cứng và nhúng vào phần giao diện của chương trình.

# PHẦN 2:PHÂN TÍCH THIẾT KẾ

## Mô hình kết nối

Sử dụng WMI để giao tiếp với công nghệ S.M.A.R.T của ổ cứng, sử dụng hệ CSDL SQLServer để lưu trữ dữ liệu.

Program

SMART

Database

WMI

SQLSERVER

Hình 1 Mô hình ứng dụng

## Cơ sở dữ liệu

SMART

Database

Table

Attribute: Các thuộc tính

RecordInfo : Thông tin bản ghi

RecordData: Dữ liệu về các bản ghi

ID

Atrribute  
Current  
Worst  
Threshold  
Data  
IsOK

Số hiệu bản ghi và giá trị thuộc tính tương ứng sẽ được lưu trữ trong bảng RecordData, bảng RecordInfo lưu trữ các thông tin về kiểu ổ cứng, số seri, ngày tháng ghi...

Hình 2 Cấu trúc cơ sở dữ liệu

## Mô hình giao diện

SMART Monitor

X

-

[]

DataGridView hiển thị các thông số về ổ cứng dưới dạng bảng, mỗi hàng là một thuộc tính

Main | Option | Report(TabControl)

Danh sách HDD (Combobox)

Thông tin cơ bản  
Serial  
Model

Trạng thái

v.v

Update

Exit

Hình 3 Giao diện chính

## Các thành phần chức năng

Dictionary<int,HDD> dDrives;

Đối tượng kiểu từ điển, đại diện cho các ổ cứng trong máy. Mỗi một ổ cứng được đánh số bằng ID, đại diện bằng một instance của class HDD.

void UpdateSMARTInfo();

Cập nhật thông tin về ổ cứng, lưu thông tin vào biến dDrives.

void ShowInfo();

Hiển thị thông tin lên giao diện chương trình.

void WriteData();

Hàm ghi dữ liệu từ dDrives vào CSDL.

Hàm được gọi khi nhấn nút Save Data. (tab Option).

void ReadData();

Hàm đọc dữ liệu từ CSDL ghi vào dDrives, được gọi khi nhấn nút Load (tab Option).

ManagementObjectSearcher("SELECT \* from Win32\_DiskDrive");

Trả về đối tượng giúp liệt kê các ổ đĩa trong máy, là phương thức khởi tạo của class ManagementObjectSearcher, có một tham số duy nhất là câu truy vấn (dạng SQL) dùng để truy vấn dữ liệu trong WMI.

ObjectQuery("SELECT \* from MSStorageDriver\_FailurePredictStatus");

ObjectQuery("SELECT \* from MSStorageDriver\_FailurePredictData");

ObjectQuery("SELECT \* from MSStorageDriver\_FailurePredictThresholds");

Trả về đối tượng hỗ trợ cho việc lấy thông tin về ổ cứng.

Câu truy vấn đầu tiên dùng để lấy thông tin dự đoán xem ổ cứng có hoạt động tốt hay không.

Câu truy vấn thứ hai dùng để lấy dữ liệu về ổ cứng, giá trị của các thuộc tính mà SMART ghi nhận được.

Câu truy vẫn thứ ba lấy giá trị Threshold của các thuộc tính.

SqlConnection conn = new SqlConnection(connectionstring)

Trả về đối tượng đại diện cho một kết nối đến SQLServer, tham số duy nhất là một chuỗi kết nối bao gồm nguồn dữ liệu, phương thức kết nối, ip/port nếu có. Ví dụ chương trình sử dụng windows authentication :

"Data Source=VGA-PC\\SQLEXPRESS;"

+ "Integrated Security=SSPI;Initial Catalog=SMARTMonitor";

.

SqlCommand cmd = new SqlCommand(string)

Tạo một đối tượng đại diện cho một truy vấn.

SqlConnection.ExecuteScalar (string Command)

Thực hiện command Insert, trả về RecordID của truy vấn.

SqlConnection.ExecuteNonQuery(String)

Thực hiện command không truy vấn (Không lấy dữ liệu về).

SqlConnection.ExecuteReader(String)

Thực hiện truy vấn tới CSDL, trả về đối tượng là con trỏ dùng để đọc dữ liệu trong bảng.

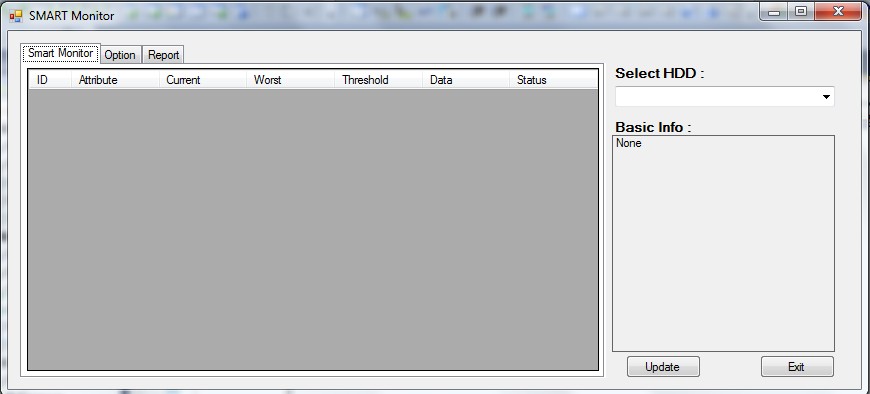
void OnStart(string[] args)

Hàm thực hiện một số công việc chỉ định khi service khởi động. Đối với chương trình này hàm OnStart được sử dụng để đọc và thiết lập thời gian giữa mỗi lần lấy thông tin.

void OnTimer(object sender, System.Timers.ElapsedEventArgs args)

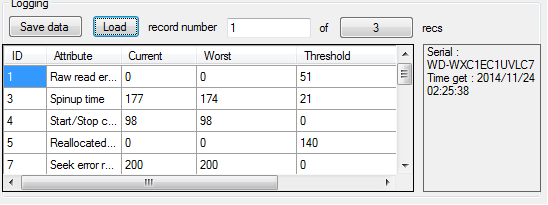
Hàm thực hiện công việc chỉ định theo thời gian trong service. Đối với chương trình này, hàm OnTimer được sử dụng để đọc thông tin và ghi vào CSDL theo thời gian.

# PHẦN 3:SẢN PHẨM KẾT QUẢ

**

Hình Giao diện chương trình

* Bấm nút update để cập nhật thông tin.
* Chọn HDD muốn xem thông tin ở phần Select HDD.



Hình Giao diện mục report

## Lưu ý

Chương trình có thể sẽ cần quyền admin để hoạt động.

Yêu cầu .Net FrameWork 3.5**.**

ReportBuilder 3.1 chỉ chạy được với SQLServer Express 2008 R2 with advanced tools trở lên.

Khi phát triển service, cần lưu ý về quyền của service. Nên sử dụng Installer để tránh phiền toái về quyền của service, không nên sử dụng installutil vì phức tạp và dễ cấp quyền sai dẫn đến việc service không hoạt động như ý.