

Министерство образования Республики Беларусь
Учреждение образования Белорусский государственный университет
информатики и радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Архитектура вычислительных систем (АВС)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту на тему

«Менеджер работы батареи»

Выполнил: студент гр.753505

Скробат Ю.А.

Руководитель: ассистент кафедры
информатики Леченко А.В.

Содержание

Введение

1. Типы аккумуляторов ноутбука

2. Используемые технологии

2.1. Среды реализации задачи

2.1.1. Microsoft Visual Studio

2.1.2. C#

2.1.3. .Net Framework

2.1.4. Windows Forms

2.1.5. Windows Management Instrumentation

2.1.6. Windows API

Заключение

Список использованных источников

Введение

Актуальность «мобильных» компьютеров. Возможность работать в любом месте и в любое удобное время стала почти необходимостью в наши дни. Производители электронной техники оперативно подхватили этот интерес и предприимчиво воплотили его в компактные элегантные мини-компьютеры. Ну разве не об этом мечталось работникам интеллектуального труда? Абсолютно все функции большой стационарной машины в таком маленьком корпусе! Неудивительно, что ноутбуки так быстро завоевали любовь пользователей и продолжают свое победное шествие по сей день.

Ноутбук – это мини-версия большого ПК. Более легкий и компактный по своим габаритам, этот аппарат уверенно занял свою нишу в мире электроники. Выполняя роль именно персональной техники, он сочетает в себе функции органайзера, mp3-плеера, фотоальбома, архива всевозможных документов, выхода в интернет, книги и развлечения.

Сильные стороны ноутбука. Безусловно, к плюсам ноутбука относятся его привлекательные параметры: вес около килограмма (иногда чуть больше), небольшие размеры самой «книжки» (как правило, диагональ монитора 10 дюймов). Клавиатура очень компактна, все близко и под рукой. Ноутбуки отличаются низким энергопотреблением, что позволяет им работать без подзарядки батареи иногда до 12 часов (как обещают производители). Мониторы обладают высоким разрешением, что обеспечивает высокое качество изображения.

1. Типы аккумуляторов у ноутбуков

Различные типы аккумуляторов в ноутбуках напрямую влияют на продолжительность автономной работы устройства. При выборе нужно учитывать не только совместимость с конкретной моделью, но и тип элемента питания, от которого зависит целый ряд свойств аккумуляторной

батареи. АКБ разных типов отличаются принципом работы и способом заряда, поэтому менять их между собой нельзя.

В современных переносных компьютерах используются следующие типы аккумуляторных батарей:

1. Литий-ионные батареи (маркировка Li-on) – самый распространенный современный тип аккумуляторов для ноутбуков. Преимуществом являются компактные размеры и небольшой вес, они также отличаются долговечностью. Литий-ионный аккумулятор рассчитан на 300 циклов полной зарядки и разрядки. Однако у него есть и важные минусы это способность к саморазряду и уменьшению емкости со временем, работать такая батарея в состоянии только при плюсовых температурах. Однако они продолжают пользоваться спросом из-за высокой производительности. Ниже представлена **схема литий-ионных аккумуляторов**:



2. Никель-кадмиевые аккумуляторы (NiCad). Продолжительность использования достигает 350 циклов зарядки и разрядки, от литий-ионных батарей они отличаются невысокой стоимостью. Главное преимущество – возможность работы в любых условиях, в том числе

при низких температурах и резких температурных перепадах. Однако есть и существенный минус – большой вес. Такие батареи относительно редко применяются в ноутбуках.

3. Никель-металл-гидридные батареи питания (NiMh). В отличие от никель-кадмиевых, они обладают увеличенной емкостью, что продлевает автономную работу ноутбука, при этом по размеру и весу они практически не отличаются. При их изготовлении не применяются тяжелые металлы поэтому они считаются наиболее безопасными экологически.
4. Литий-полимерные аккумуляторы (Li-polymer) - относительно новое изобретение, обладающее маленькими габаритными размерами. Такие элементы питания используются в различных небольших гаджетах, их можно справедливо назвать технологией будущего. Одно из преимуществ – возможность придания любой формы, что дополнительно расширяет возможности использования в портативных устройствах.

3. Используемые технологии

Приложение реализовано на языках C# с использованием технологий ASP.NET Framework, Windows Forms, WMI(Windows Management Instrumentation), Windows API. В качестве среды разработки была выбрана Microsoft Visual Studio 2017.

2.1. Среды реализации задачи

2.1.1. Microsoft Visual Studio

Microsoft Visual Studio – это новая разработка компании Microsoft, позволяющая создавать приложения, работающие на платформе .NET.

Особенность этой платформы заключается в широком наборе сервисов, которые доступны в различных языках программирования. При этом сервисы реализуются в виде промежуточного кода, который не зависит от базовой

архитектуры. Едва ли не главной целью создания такой платформы было оснащение разработчиков специальными сервисно-ориентированными приложениями, которые могли бы работать на любой платформе, начиная от персонального компьютера и заканчивая мобильным устройством.

Microsoft Visual Studio объединяет в себе огромное количество функций, позволяющих осуществлять разработки для Windows всех версий, в том числе и Windows 8, интернета, SharePoint, различных мобильных устройств и облачных технологий. В Visual Studio реализуется новая среда разработчика, благодаря которой создавать приложения стало проще.

Microsoft Visual Studio – это обновленная и упрощенная программная среда, для которой характерна высокая производительность, причем она не зависит от особенностей оборудования.

2.1.2. C#

C# разрабатывался как язык программирования прикладного уровня для CLR и, как таковой, зависит, прежде всего, от возможностей самой CLR. Это касается, прежде всего, системы типов C#, которая отражает BCL. Присутствие или отсутствие тех или иных выразительных особенностей языка диктуется тем, может ли конкретная языковая особенность быть транслирована в соответствующие конструкции CLR. Так, с развитием CLR от версии 1.1 к 2.0 значительно обогатился и сам C#; подобного взаимодействия следует ожидать и в дальнейшем. (Однако эта закономерность была нарушена с выходом C# 3.0, представляющего собой расширения языка, не опирающиеся на расширения платформы .NET.) CLR предоставляет C#, как и всем другим .NET-ориентированным языкам, многие возможности, которых лишены «классические» языки программирования. Например, сборка мусора не реализована в самом C#, а производится CLR для программ, написанных на C# точно так же, как это делается для программ на VB.NET, J# и др.

3.1.3. .Net Framework

Платформа .NET Framework — это технология, которая поддерживает создание и выполнение нового поколения приложений и веб-служб XML. При разработке платформы .NET Framework учитывались следующие цели.

- Обеспечение согласованной объектно-ориентированной среды программирования для локального сохранения и выполнения объектного кода, для локального выполнения кода, распределенного в Интернете, либо для удаленного выполнения.
- Обеспечение среды выполнения кода, минимизирующей конфликты при развертывании программного обеспечения и управлении версиями.
- Обеспечение среды выполнения кода, гарантирующей безопасное выполнение кода, включая код, созданный неизвестным или не полностью доверенным сторонним изготовителем.
- Обеспечение среды выполнения кода, исключающей проблемы с производительностью сред выполнения сценариев или интерпретируемого кода.
- Обеспечение единых принципов разработки для разных типов приложений, таких как приложения Windows и веб-приложения.
- Взаимодействие на основе промышленных стандартов, которое гарантирует интеграцию кода платформы .NET Framework с любым другим кодом.

2.1.4. Windows Forms

Windows Forms — интерфейс программирования приложений (API), отвечающий за графический интерфейс пользователя и являющийся частью Microsoft .NET Framework. Данный интерфейс упрощает доступ к элементам интерфейса Microsoft Windows за счет создания обёртки для

существующего Win32 API в управляемом коде. Причём управляемый код — классы, реализующие API для Windows Forms, не зависят от языка разработки. То есть программист одинаково может использовать Windows Forms как при написании ПО на C#, C++, так и на VB.Net, J# и др.

С одной стороны, Windows Forms рассматривается как замена более старой и сложной библиотеке MFC, изначально написанной на языке C++. С другой стороны, WF не предлагает парадигму, сравнимую с MVC. Для исправления этой ситуации и реализации данной функциональности в WF существуют сторонние библиотеки. Одной из наиболее используемых подобных библиотек является User, Interface, Process, Application, Block, выпущенная специальной группой Microsoft, занимающейся примерами и рекомендациями, для бесплатного скачивания. Эта библиотека также содержит исходный код и обучающие примеры для ускорения обучения.

2.1.5 Windows Management Instrumentation

Windows Management Instrumentation (WMI) в дословном переводе — инструментарий управления Windows. WMI — это одна из базовых технологий для централизованного управления и слежения за работой различных частей компьютерной инфраструктуры под управлением платформы Windows.

2.1.6. Windows API

Windows API спроектирован для использования в языке Си для написания прикладных программ, предназначенных для работы под управлением операционной системы MS Windows. Работа через Windows API — это наиболее близкий к операционной системе способ взаимодействия с ней из прикладных программ. Более низкий уровень доступа, необходимый только для драйверов устройств, в текущих версиях Windows предоставляется через Windows Driver Model.

Заключение

В рамках работы над курсовым проектом был разработан программный продукт для управления работой батареи.

За время написания проекта была изучена информация о принципах работы батареи и подходы решения основных задач, возлагаемых на профсоюзную организацию.

В плане разработки приложения был изучен большой объем информации по проектированию приложения, углублены и закреплены знания по разработке приложений на платформе .NET Framework.

Были проанализированы различные подходы и технологии, из которых были выбраны те, которые автор хотел бы видеть в своем программном продукте.

Разработанное программное средство представляет собой законченный программный продукт, готовый к использованию. Но при желании приложение можно доработать: расширить функциональность приложения, изменить дизайн и др.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

https://ru.wikipedia.org/wiki/Windows_Forms

<https://ru.wikipedia.org/wiki/WMI>

<https://docs.microsoft.com/ru-ru/dotnet/framework/get-started/overview>

<https://vnoutbuke.ru/faq/tipy-akkumulyatorov-u-noutbukov/>