Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Архитектура вычислительных систем

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

на тему

ВИДЖЕТ МОНИТОРИНГА СОСТОЯНИЯ АППАРАТНОГО ОБЕСПЕЧЕНИЯ КОМПЬЮТЕРА И ПРОГНОЗА ВРЕМЕНИ ЕГО РАБОТЫ

БГУИР КП 1-40 04 01 01 094

Студент Хрищанович А.К.

Руководитель Марков А.Н.

Минск 2023

СОДЕРЖАНИЕ

[Введение 3](#_Toc147187568)

[1 Архитектура вычислительной системы 4](#_Toc147187569)

[1.1 Структура и архитектура вычислительной системы 4](#_Toc147187570)

[1.2 История, версии и достоинства 9](#_Toc147187571)

[1.3 Анализ вычислительной системы и обоснование выбора для написания программы 13](#_Toc147187572)

[2 Платформа программного обеспечения 15](#_Toc147187573)

[2.1 Структура и архитектура платформы 15](#_Toc147187574)

[2.2 История, версии и достоинства 17](#_Toc147187575)

[2.3 Анализ программного обеспечения для написания программы и обоснование выбора платформы 19](#_Toc147187576)

[Приложение А (обязательное) Листинг кода 22](#_Toc147187577)

# **ВВЕДЕНИЕ**

В современном мире компьютеры играют неотъемлемую роль в повседневной жизни. Они используются в различных сферах, начиная от домашних рабочих станций и заканчивая критически важными системами, такими как медицинские системы и серверы данных. Поэтому актуальность темы «Виджет мониторинга состояния аппаратного обеспечения компьютера и прогноза времени его работы» трудно переоценить, что означает, что данная тема имеет высокую актуальность, так как связана с повседневной жизнью многих людей и организаций, которые зависят от надежной работы своих технических средств.

Цель данного курсового проекта заключается в разработке и реализации виджета мониторинга состояния аппаратного обеспечения компьютера, который способен предоставить пользователю информацию о текущем состоянии компьютера, а также прогнозировать оставшееся время работы компьютера. Для достижения данной цели были поставлены следующие задачи:

– Анализ существующих решений: провести обзор существующих программных средств и решений для мониторинга аппаратного обеспечения компьютера и определить их преимущества и недостатки.

– Разработка виджета мониторинга состояния аппаратного обеспечения компьютера: разработать программный виджет, который будет собирать информацию о состоянии различных аппаратных компонентов компьютера, таких как процессор, оперативная память, жесткий диск и видеокарта.

– Прогнозирование работы компьютера: разработать алгоритм прогнозирования работы компьютера на основе данных, собранных виджетом мониторинга. Это позволит получать информацию о возможных проблемах и неисправностях заранее.

– Интерфейс пользователя: создать удобный и информативный интерфейс для виджета, который будет понятен и доступен для широкого круга пользователей.

Для достижения цели курсовой работы, необходимо разработать и реализовать виджет мониторинга состояния аппаратного обеспечения компьютера, который будет собирать и анализировать данные о состоянии компьютера, а также прогнозировать его работоспособность. Для этого будут использоваться методы анализа данных и разработки программного обеспечения.

Таким образом курсовая работа направлена на повышения уровня информированности пользователя о состоянии своих компьютеров, что позволит им своевременно предотвращать возможные проблемы и сбои в работе, а также улучшить общую производительность и долговечность аппаратного обеспечения компьютера.

# **1 АРХИТЕКТУРА ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ**

## **Структура и архитектура вычислительной системы**

Вычислительная система (ВС) – результат интеграции аппаратных и программных средства, функционирующих в единой системе и предназначенных для решения задач определенного класса. Структура вычислительной системы состоит из пяти уровней:

– Аппаратный уровень вычислительной системы: определяется набором аппаратных компонентов и их характеристиками, используемых вышестоящими уровнями иерархии и определяющими воздействие на них. К физическим ресурсам этого уровня относятся: процессор, оперативная память, внешние устройства.

[1, 15 c.]

– Уровень управления физическими ресурсами вычислительной системы: уровень управления физическими ресурсами ВС является первым уровнем системного программного обеспечения вычислительной системы, его назначение – систематизация и стандартизация правил программного использования физических ресурсов. На этом уровне обеспечивается создание программ управления физическими ресурсами. Для обеспечения управления физическими ресурсами используются программы, которые называются драйверами физического ресурса (устройства).

Драйвер физического устройства – программа, основанная на использовании команд управления конкретного физического устройства и предназначенная для организации работы с данным устройством.

– Уровень управления логическими/виртуальными ресурсами вычислительной системы:

Логическое/виртуальное устройство (ресурс) – устройство, некоторые эксплуатационные характеристики которого (возможно, все) реализованы программным образом.

Драйвер логического/виртуального ресурса – программа, обеспечивающая существование и использование соответствующего ресурса.

Этот уровень ориентирован на пользователя. Команды данного уровня не зависят от физических устройств, они обращены к предыдущему уровню. На базе этого уровня могут создаваться новые логические ресурсы. При организации драйвера логического устройства могут использоваться драйверы физических или логических/виртуальных устройств. Система поддерживает иерархию драйверов. Многоуровневая унификация интерфейса.

Ресурсы вычислительной системы – совокупность всех физических и виртуальных ресурсов. Одной из характеристик ресурсов является их конечность, следовательно, возникает конкуренция за обладание ресурсом между его программными потребителями.

Операционная система – это комплекс программ, обеспечивающий управление ресурсами вычислительной системы.

Разветвленная иерархия виртуальных и физических устройств. Драйверы можно разделить на 3 группы:

Драйверы физических устройств

Драйверы виртуальных устройств, обобщающих характеристики соответствующих физических устройств

Драйверы виртуальных устройств, не имеющих аппаратной реализации (т.е. "полностью" виртуальных, например, драйвер файловой системы).

– Уровень систем программирования:

Система программирования – это комплекс программ, обеспечивающий поддержание жизненного цикла программы в вычислительной системе.

Жизненный цикл программы в вычислительной системе состоит из четырех основных этапов:

– Проектирование программного продукта.

– Кодирование (программная реализация).

– Тестирование и отладка. Тестирование – проверка программы на тестовых нагрузках. Тест – заранее определенные входные данные. Тестовое покрытие – минимальный набор тестов, наиболее полно проверяющих программу.

Отладка – процесс поиска, локализации и исправления зафиксированных при тестировании ошибок.

Внедрение и сопровождение. Внедрение – установка программного комплекса на объектную вычислительную систему и его первичная настройка. Сопровождение – исправление недочетов разработки и внедрения программного комплекса (например, выпуск патчей).

– Уровень прикладных систем

Прикладная система – программная система, ориентированная на решение или автоматизацию решения задач из конкретной предметной области.

[1, 15-17 c.]

Цифровой компьютер состоит из связанных между собой процессоров, модулей памяти и устройств ввода-вывода.

**1.1.1** Выбранная вычислительная система

Вычислительная система, на которой проводится разработка данного курсового проекта, базируется на операционной системе Windows 10 и аппаратном обеспечении ноутбука Asus из серии TUF Gaming. Эта система включает в себя следующие компоненты:

– Процессор Intel Core i5: процессор (центральный процессор) – основной вычислительный движок компьютера. Обеспечивает высокую прооизводительность и является важной частью архитектуры системы.

– Оперативная память (RAM): оперативная память служит для временного хранения данных, используемых программами во время их выполнения. Это важный аспект архитектуры, так как от объемы и скорости оперативной памяти зависит производительность системы.

– Жестки диск: жесткий диск или SSD предназначен для хранения операционной системы, приложений и данных.

– Видеокарта: ноут оснащен дискретной видеокартой, которая обеспечивает высокую производительность в графических приложениях и играх.

– Процессор NVIDIA GeForce GTX 1650, Intel® UHD Graphics

**1.1.2** Устройство центрального процессора

Процессор – мозг компьютера. Его задача – выполнять программы, находящиеся в основной памяти. Для этого он вызывает команды из памяти, определяет их тип, а затем выполняет одну за другой. Компоненты соединены шиной, представляющей собой набор параллельно связанных проводов для передачи адресов, данных и управляющих сигналов.

Процессор состоит из нескольких частей. Блок управления отвечает за вызов команд из памяти и определение их типа. Арифметика-логическое устройство выполняет арифметические операции и логические операции.

[2, 77 c.]

Трак данных состоит из регистров (обычно от 1 до 32), арифметико-логического устройства (АЛУ) и нескольких соединительных шин. Содержимое регистров поступает во входные регистры АЛУ. В них находятся входные данные АЛУ, пока АЛУ производит вычисления. Тракт данных – важная составляющая часть всех компьютеров.

[2, 78 c.]

Процессор – это только один из компонентов компьютера. В дополнении к нему требуются по крайней мере некоторое оперативное запоминающее устройство (ОЗУ), какой-то способ, позволяющий пользователю записать информацию в компьютер (устройство ввода), а также извлечь ее из него (устройство вывода). Кроме того, необходимо еще несколько микросхем для объединения всех компонентов. Сам микропроцессор мы можем представить в качестве «черного ящика», внутреннюю работу которого нам не обязательно досконально изучать, чтобы разобраться в его функциях. Для этого достаточно ознакомиться с входными и выходными сигналами, в частности с набором команд.

[3, 476 c.]

Центральный процессор выполняет каждую команду за несколько шагов. Он делает следующее:

– Вызывает следующую команду из памяти и переносит ее в регистр команд.

– Меняет положение счетчика команд, который после этого указывает на следующую команду.

– Определяет тип вызванной команды.

– Если команда использует слово из памяти, определяет, где находится это слово.

– Переносит слово, если это необходимо, в регистр центрального процессора.

– Выполняет команду.

– Переходит к первому шагу, чтобы начать выполнение следующей команды.

Такая последовательность шагов (выборка – декодирование – исполнение) является основой работы всех компьютеров.

[2, 79 c.]

**1.1.3** Основная (оперативная) память

В общем случае основной памятью компьютера называют оперативную память (Random Access Memory, RAM). Оперативная память играет ключевую роль в функционировании компьютера и используется для временного хранения данных и кода, которые активно используются программами во время их выполнения.

Память – это тот компонент компьютера, в котором хранятся программы и данные. Также часто встречается термин «запоминающее устройство». Без памяти, откуда процессоры считывают и куда записывают информацию, не было бы современных цифровых компьютеров.

Основной единицей измерения хранения данных в памяти является двоичный разряд, который называется битом. Бит может содержать 0 или 1. Эта самая маленькая единица памяти.

Применение двоичной системы счисления в компьютерах объясняется ее «эффективностью». При этом имеется в виду, что хранение цифровой информации может быть основано на отличиях между разными величинами какой-либо физической характеристики, например напряжения или тока. Чем больше величин нужно различать, тем меньше отличий между смежными величинами и тем менее надежна память. В двоичной системе счисления требуется различать всего две величины, следовательно, это – самый надежный метод кодирований цифровой информации.

[2, 94 c.]

Память состоит из ячеек, каждая из которых может хранить некоторую порцию информации. Каждая ячейка имеет номер, который называется адресом. По адресу программы могут ссылаться на определенную ячейку.

[2, 95 c.]

В компьютерах, в которых используется двоичная система счисления (включая восьмеричной и шестнадцатеричное представление двоичных числе), адреса памяти также выражаются в двоичных числах.

Ячейка – минимальная адресуемая единица памяти. В последние годы практически все производители выпускают компьютеры с 8-разрядными ячейками, которые называются байтами. Байты группируются в слова.

[2, 96 c.]

**1.1.4** Вспомогательная память

Каков бы ни был объем основной памяти, ее все равно будет мало. С развитием технологий людям приходят в голову такие вещи, которые раньше считались совершенно фантастическими. Огромный объем информации в настоящее время невозможно разместить в основной памяти, поэтому на данный момент все пользуются вспомогательной памятью.

Устройства на базе энергонезависимой флэш-памяти, часто называют твердотельными накопителями или SSD-дисками (Solid State Disk), постепенно начинают рассматриваться как высокоскоростная альтернативна традиционным технологиям магнитных дисков.

[2, 119 c.]

Так как SSD-диски по сути являются памятью, они обладают более высокой производительностью по сравнению с вращающимися магнитными дисками при нулевом времени поиска. Поскольку устройство не имеет подвижных частей, оно особенно хорошо подходит для ноутбуков (колебания в перемещении не влияют на его способность обращаться к данным).

[2, 121 c.]

**1.1.5** Графический процессор

Несмотря, что набор инструкций доступный для программирования GPU проще, чем для CPU, содержание этих инструкций также в точности не соответствует действиям оборудования. Причина кроется в том, что мы не можем просто запрограммировать графический процессор без какого-либо API, который является абстракцией над его внутренней работой.

Давайте перечислим наиболее важные части GPU на самом низком из возможных уровней: Блоки FP32 FP 64 и/или SPU (специализированные вычислительные блоки) Блоки INT32 Регистры встроенная память: L0$ (редко в оборудовании потребительского класса) L1$ используется совместно в разделяемой памятью (Shared Memory) на каждом вычислительном блоке L2$ на GPU внешняя память (память устройства) планировщик команд диспетчер команд TMU, ROP — фиксированные блоки конвейера …и другие. Все вышеперечисленное, конечно, сгруппировано во множество иерархий, которые в итоге составляют графические карты.

GPU — это одно большое устройство асинхронного ожидания. По крайней мере, так вы можете думать о нем с точки зрения CPU. Конечно, это намного сложнее, чем все доступное для взгляда со стороны процессора. Но все же логика остается. Вы отправляете какую-то работу и ждете результатов, занимаясь другими делами. Вам нужно обеспечить какую-то синхронизацию сделанного и несделанного.

Ядра CUDA в картах Nvidia или просто ядра в графических процессорах AMD — это очень простые блоки, которые выполняют операции конкретно с плавающей точкой. Они не могут выполнять такие же причудливые вещи, как процессоры (например, прогнозирование ветвлений, выполнение вне очереди, выборку данных). В то же время они не могут работать самостоятельно.

[8]

## **История, версии и достоинства**

**1.2.1** Архитектура x64

Архитектура x86-64, также известная как Intel 64, представляет собой расширение архитектуры x86, которая была разработана компаниями AMD и Intel. Ее история связана с потребностью в поддержке больших объемов памяти и более высокой производительности. Она была представлена в 2000 году.

С тех пор архитектура x86-64 пережила несколько версий и улучшений. Наиболее значимыми версиями были:

– AMD64: исходная версия архитектуры, представленная AMD в 2000 году.

– Intel 64: Intel также внедрила свою версию архитектуры x86-64, известную как Intel 64, также в начале 2000-х годов.

Архитектура x86-64 имеет ряд значительных преимуществ:

– Поддержка 64-битных приложений: позволяет запускать и обрабатывать 64-битные приложения, что повышает производительность и позволяет эффективно управлять большими объемами данных.

– Поддержка больших объемов памяти: способность адресации более 4 ГБ оперативной памяти, что является критически важным для современных вычислительных задач.

– Совместимость: обратная совместимость с 32-битными приложениями и операционными системами, что обеспечивает плавный переход к 64-разрядной архитектуре.

Архитектура x86-64 внесла множество инноваций и технологических решений, включая:

– Наборы инструкций: расширенные наборы инструкций, которые обеспечивают более высокую производительность и возможности оптимизации.

– Виртуализация: встроенная поддержка технологий виртуализации для лучшей изоляции и управления виртуальными машинами.

– Защита данных: механизмы защиты данных и аппаратная защита от вредоносных атак.

Архитектура x86-64, в том числе Intel 64, широко применяется в различных областях:

– Серверы: в крупных серверных фермах для обработки данных и виртуализации.

– Настольные и ноутбуки: в персональных компьютерах для выполнения разнообразных задач, включая игры, мультимедиа и офисные приложения.

– Разработка и научные исследования: для высокопроизводительных вычислений и моделирования.

Архитектура x86-64 соперничает с другими архитектурами, такими как ARM64, в различных областях, и каждая из них имеет свои сильные стороны и применение. Сравнительный анализ может зависеть от конкретных задач и требований.

**1.2.2** Оперативная память (RAM) 8 ГБ

История развития оперативной памяти в компьютерах имеет долгий путь, начиная с ранних электромеханических устройств и электронных трубок в середине 20-го века. С появлением первых компьютеров, оперативная память представляла собой регистры и буферы, которые использовались для хранения ограниченного количества данных. Со временем разработчики стали осознавать необходимость увеличения объема и скорости оперативной памяти.

Первые интегральные схемы для оперативной памяти появились в 1970-х годах, что позволило значительно увеличить ее объем и скорость. Затем пришли стандарты памяти, такие как DRAM (динамическая оперативная память) и SRAM (статическая оперативная память), которые сделали оперативную память более доступной и эффективной.

С появлением компьютеров с графическими пользовательскими интерфейсами в 1980-х и 1990-х годах, объем оперативной памяти стал критически важным для обеспечения плавной работы графических приложений и многозадачности.

В последние десятилетия, с развитием многоядерных процессоров и 64-разрядных операционных систем, объем и скорость оперативной памяти продолжают расти. Современные стандарты памяти DDR3, DDR4 и DDR5 предлагают высокую производительность и эффективность.

На протяжении истории компьютерных технологий существует множество версий оперативной памяти. Однако основные различия заключаются в следующем:

– DDR (Double Data Rate) – первые стандарты, которые существуют с 2000 года.

– DDR2 – следующее поколение памяти, улучшившее скорость передачи данных.

– DDR3 – стандарт, который сделал память еще быстрее и эффективнее.

– DDR4 – последнее поколение на момент написания, предлагающее высокую производительность и низкое энергопотребление.

– DDR5 – ожидаемый стандарт, который будет продолжать увеличивать производительность.

Оперативная память играет ключевую роль в производительности компьютерных систем. Ее основные преимущества включают в себя:

– Быстрый доступ к данным, что ускоряет выполнение задач.

– Поддержка многозадачности, позволяя одновременно выполнять несколько приложений.

– Улучшение производительности в графических и видео приложениях.

– Снижение времени загрузки операционной системы и приложений.

– Надежность и стабильность в работе системы.

Инновации в оперативной памяти включают в себя управление энергопотреблением для экономии электроэнергии, технологии исправления ошибок для обеспечения надежности данных и увеличение скорости передачи данных с каждым новым стандартом.

Оперативная память сравнивается с альтернативными формами хранения данных, такими как жесткие диски и SSD (твердотельные накопители). В сравнении с ними, оперативная память предлагает мгновенный доступ к данным, что делает ее идеальным выбором для выполнения задач, требующих высокой скорости чтения и записи.

Оперативная память широко используется во всех областях информационных технологий, включая домашние компьютеры, серверы, мобильные устройства и встраиваемые системы. Ее выдающиеся задачи включают в себя поддержку операционных систем, обработку данных в реальном времени и обеспечение высокой производительности в играх и профессиональных приложениях.

**1.2.3** SSD-диск Micron 2210 512 ГБ

Микрон (Micron Technology) – это американская компания, специализирующаяся на производстве полупроводников и твердотельных накопителей. Что касается SSD дисков Micron, их история связана с развитием технологии NAND-флэш памяти и ростом рынка твердотельных накопителей. Микрон начала производство SSD дисков в начале 2010-х годов. С течением времени SSD диски Micron стали обладать большей емкостью, более высокой скоростью чтения и записи, а также улучшенной надежностью благодаря развитию NAND-технологий и контроллеров. Рост объемов данных, требования к скорости и надежности, а также снижение стоимости NAND-флэш памяти оказали влияние на эволюцию SSD дисков Micron.

Серия SSD дисков Micron 2210, как правило, имеет несколько версий с различными объемами памяти и характеристиками скорости. Улучшения могут включать в себя увеличение емкости, повышение скорости чтения/записи и снижение энергопотребления.

К достоинствам SSD-дисков Micron можно отнести:

– Высокая скорость чтения и записи данных.

– Надежность и долгий срок службы.

– Отсутствие подвижных деталей, что делает их устойчивыми к физическим ударам.

– Относительно низкое энергопотребление.

– Улучшение производительности системы благодаря быстрому доступу к данным.

Micron внедряет новые технологии NAND-флэш памяти, такие как 3D NAND, и совершенствует контроллеры, чтобы улучшить производительность и надежность своих SSD дисков.

SSD диски Micron 2210 применяются в различных областях, включая настольные компьютеры, ноутбуки, серверы и встраиваемые системы. Они эффективно работают в задачах, где требуется быстрый доступ к данным, надежность и производительность, таких как загрузка операционной системы, работа с приложениями и обработка данных.

**1.2.4** Видеокарты NVIDIA GeForce GTX 1650 и Intel(R) UHD Graphics

Архитектура графических процессоров (GPU) развивалась на протяжении многих лет. Первые GPU были разработаны в конце 20-го века и предназначались для ускорения графических вычислений на компьютерах. С развитием видеоигр и требований к графике, графические процессоры стали более мощными и функциональными.

Создание дискретных видеокарт, таких как NVIDIA GeForce GTX 1650, связано с ростом популярности компьютерных игр и профессиональных графических приложений. Даты выпуска различных версий видеокарт и ключевые изменения в их архитектуре также являются важными моментами в истории.

В архитектуре GPU NVIDIA GeForce GTX есть множество версий и поколений. Каждое новое поколение обычно вносит улучшения в производительность, эффективность и функциональность. Примерами могут служить различные модели в серии GeForce GTX, такие как 900-я, 1000-я, и 1600-я серии. Важно рассмотреть, какие конкретные изменения и улучшения были внесены в каждую версию и почему они были важны для пользователей.

Архитектура NVIDIA GeForce GTX 1650 обладает рядом преимуществ, включая высокую графическую производительность, поддержку современных графических технологий, таких как Ray Tracing, эффективное охлаждение, и возможность использования в игровых и профессиональных приложениях. Эти достоинства могут существенно повысить производительность и качество визуальных задач на компьютере.

В архитектуре GPU всегда присутствуют инновации и технологии. Это могут быть новые алгоритмы обработки графики, улучшенные методы рендеринга, аппаратная поддержка искусственного интеллекта, и многое другое.

## **1.3 Анализ вычислительной системы и обоснование выбора для написания программы**

В первую очередь, основной причиной выбора данной вычислительной системы, описанной в пункте 1.2, стало наличие ноутбука, основанного на данной вычислительной системе. Присутствие уже имеющегося устройства на руках сэкономило время, которое могло бы быть затрачено на поиск и приобретение необходимого оборудования.

Кроме того, при выборе вычислительной системы важным фактором было обеспечение комфортных условий для работы, создания виджета и взаимодействия с аппаратным обеспечением. В данном случае выбранная вычислительная система отвечает этим требованиям по нескольким причинам:

– Производительность процессора Intel Core i5: процессор Intel Core i5 предоставляет хорошую производительность для выполнения широкого спектра задач. Его многозадачные способности позволяют комфортно работать с различными приложениями, включая разработку программного обеспечения, обработку данных и другие вычислительно интенсивные задачи.

– Объем и скорость оперативной памяти (RAM): выбор оперативной памяти является важным аспектом архитектуры системы. 8 ГБ оперативной памяти, предоставляемой этой системой, обеспечивают достаточное количество ресурсов для эффективной работы с приложениями, включая среды разработки и виртуальные машины.

– Скорость и емкость жесткого диска (SSD): присутствие SSD обеспечивает высокую скорость загрузки операционной системы и приложений, что улучшает общую производительность системы. Он также обеспечивает хорошую отзывчивость и быстрое открытие файлов.

– Дискретная видеокарта NVIDIA GeForce GTX 1650: для выполнения графических задач и игр, а также для разработки и тестирования графических приложений, важна дискретная видеокарта. NVIDIA GeForce GTX 1650 обеспечивает высокую производительность и качественное отображение графики.

– Совместимость с программными средами: Windows 10 является популярной операционной системой, обеспечивающей совместимость с большим количеством программ и сред разработки. Это важно для выполнения курсового проекта и обеспечения легкости в разработке и отладке программ.

– Портативность и мобильность: ноутбук Asus из серии TUF Gaming предоставляет портативность, что позволяет работать над проектом в различных местах и условиях, что может быть важно для эффективного выполнения задач.

В целом, выбранная вычислительная система обеспечивает баланс между производительностью, совместимостью, портативностью и доступностью, что делает ее максимально оптимальным решением для выполнения данного курсового проекта.

# **2 ПЛАТФОРМА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**2.1** **Структура и архитектура платформы**

**2.1.1** Выбранная платформа

Разработка проекта «Виджет мониторинга состояния аппаратного обеспечения компьютера и прогноза времени его работы» ведется на языке Python с использованием среды разработки PyCharm.

Основной платформой разработки виджета является Python, так как именно этот язык программирования был выбран для проектирования проекта. [ОПИСАНИЕ БИБЛИОТЕК, ЭТО ПОПОЗЖЕ, ПОКА Я НЕ РАЗРАБАТЫВАЛА ПРОГРАММУ].

PyCharm – это специализированная интегрированная среда разработки Python, предоставляющая широкий спектр необходимых инструментов для разработчиков, тесно интегрированных для создания удобной среды для продуктивной разработки. Все, что происходит в PyCharm, происходит в контексте проекта. Он служит основой для помощи в кодировании, массового рефакторинга, согласованности стилей кодирования и прочего.

[9]

Для работы с проектом была выбрана операционная система Windows. Windows – это операционная система, созданная корпорацией Microsoft. Операционная система (ОС) – главная программа, которая запускается при включении компьютера. Она позволяет пользователям компьютера работать с файлами, пользоваться нужными приложениями, выходить в Интернет.

[10]

**2.1.2** Язык программирования

Язык программирования – формальный язык, предназначенный для записи компьютерных программ. Язык программирования определяет набор лексических, синтактических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель под ее управлением.

Язык программирования предназначен для написания компьютерных программ, которые представляют собой набор правил, позволяющих компьютеру выполнить тот или иной вычислительный процесс, организовать управление различными объектами и прочее. Язык программирования отличается от естественных языков тем, что предназначен для управления ЭВМ, в то время как естественные языки используются, прежде всего, для общения людей между собой. Большинство языков программирования использует специальные конструкции для определения и манипулирования структурами данных и управления процессом вычислений.

**2.1.3** Интегрированная среда разработки

Интегрированная среда разработки (IDE) – комплекс программных средств, используемый программистами для разработки программного обеспечения.

Среда разработки включает в себя:

– Текстовый редактор;

– Транслятор (компилятор или интерпретатор);

– Средства сборки и отладки.

Интегрированные среды разработки были созданы для того, чтобы максимизировать производительность программиста благодаря тесно связанным компонентам с простыми пользовательскими интерфейсами.

ИСР обычно представляет собой единственную программу, в которой проводится вся разработка. Она, как правило, содержит много функций для создания, изменения, компилирования, развертывания и отладки программного обеспечения. Цель интегрированной среды заключается в том, чтобы объединить различные утилиты в одном модуле, который позволит абстрагироваться от выполнения вспомогательных задач, тем самым позволяя программисту сосредоточиться на решении собственно алгоритмической задачи и избежать потерь времени при выполнении типичных технических действий (например, вызове компилятора). Таким образом, повышается производительность труда разработчика. Также считается, что тесная интеграция задач разработки может далее повысить производительность за счёт возможности введения дополнительных функций на промежуточных этапах работы. Например, ИСР позволяет проанализировать код и тем самым обеспечить мгновенную обратную связь и уведомить о синтаксических ошибках.

**2.1.4** Уровень операционной системы

Как уже отмечалось, современный компьютер организован в виде иерархии уровней, каждый из которых добавляет определенные функции к нижележащему уровню.

С точки зрения программиста, операционная система – это программа, добавляющая ряд команд и функций к командам и функциям, предлагаемым уровнем архитектуры команд. Обычно операционная система организуется программно, но нет никаких веских причин, по которым ее, как микропрограммы, нельзя было реализовать.

Все команды уровня операционной системы доступны

для прикладных программистов. Это — практически все команды более низкого уровня, а также новые команды, добавленные операционной системой. Новые команды называются системными вызовами. Они вызывают предопределенную службу операционной системы, в частности одну из ее команд. Например, типичный системный вызов может читать данные из файла.

[2, 475 c.]

**2.2 История, версии и достоинства**

**2.2.1** Язык программирования Python

Python - это мощный и популярный язык программирования, который был создан Гвидо ван Россумом и впервые выпущен в конце 1980-х годов.

Работа над Python началась в конце 1980-х годов, и первая публичная версия, Python 0.9.0, была выпущена в 1991 году. Основными целями создания Python были читаемость кода, простота и ясность.

Версии Python 2.x стали очень популярными и использовались в различных проектах. Однако, с развитием языка возникли некоторые проблемы, связанные с совместимостью и дизайном.

Python 3.x (2008-н.в.): Python 3.x был выпущен в 2008 году и представил множество улучшений и изменений, направленных на устранение проблем, обнаруженных в Python 2.x. Это привело к некоторым разрывам совместимости, но с течением времени многие проекты перешли на Python 3.x.

Python имеет множество версий, но две основные ветки - Python 2.x и Python 3.x. Каждая из этих веток имеет свои подверсии, и каждая версия вносит улучшения и изменения. На момент окончания моего знания (сентябрь 2021 года), актуальной версией Python была Python 3.9. Кроме того, Python активно развивается, и новые версии регулярно выпускаются с улучшениями и новыми возможностями.

Python пользуется широкой популярностью благодаря ряду выдающихся достоинств:

- Простота и читаемость: Python известен своей простотой и легкочитаемым синтаксисом, который делает код более понятным и поддерживаемым.

- Множество библиотек и фреймворков: Python имеет богатую экосистему библиотек и фреймворков для разработки, что упрощает создание разнообразных приложений.

- Кроссплатформенность: Python поддерживает множество операционных систем, что делает его доступным на различных платформах.

- Широкое применение: Python применяется в разных областях, включая веб-разработку, научные вычисления, искусственный интеллект, анализ данных и многое другое.

- Активное сообщество: Python имеет активное сообщество разработчиков и пользователей, что обеспечивает поддержку и развитие языка.

**2.2.2** Среда разработки PyCharm

PyCharm - это одна из самых популярных сред разработки для языка программирования Python. Она была создана компанией JetBrains и впервые выпущена в 2010 году. Вот ключевые моменты в истории PyCharm:

Работа над PyCharm началась в JetBrains как ответ на растущую популярность языка Python. Целью было создание мощной и интуитивно понятной среды разработки для Python-разработчиков.

Первая версия PyCharm была выпущена в феврале 2010 года. Она предоставила множество возможностей, таких как интегрированный отладчик, автодополнение кода, управление виртуальными окружениями и многое другое.

С течением времени PyCharm стала одной из самых популярных сред разработки Python и продолжает активно развиваться и совершенствоваться.

PyCharm имеет несколько версий, каждая из которых предоставляет разные уровни функциональности и целевые аудитории. На момент окончания моего знания (сентябрь 2021 года), существовали следующие версии PyCharm:

- PyCharm Community Edition: Бесплатная версия PyCharm, предназначенная для небольших проектов и индивидуальных разработчиков.

- PyCharm Professional Edition: Платная версия PyCharm, предоставляющая расширенные возможности для профессиональных разработчиков и команд.

PyCharm пользуется выдающимися достоинствами, которые делают ее одной из предпочтительных сред разработки Python:

- Мощный редактор кода: PyCharm обладает мощным и интуитивно понятным редактором кода с функциями автодополнения, подсветки синтаксиса и другими инструментами для удобной работы с кодом.

- Интегрированный отладчик: Встроенный отладчик позволяет проще и быстрее находить и устранять ошибки в коде.

- Управление виртуальными окружениями: PyCharm облегчает создание и управление виртуальными окружениями, что полезно при работе с разными проектами и библиотеками.

- Поддержка множества фреймворков: PyCharm поддерживает различные фреймворки, включая Django, Flask, PyQT и другие, упрощая разработку веб-приложений и других приложений.

- Анализ кода и рефакторинг: Среда предоставляет множество инструментов для анализа кода и проведения рефакторинга, что помогает улучшить качество кода.

**2.2.3** Операционная система Windows

Операционная система Windows разработана корпорацией Microsoft и является одной из самых распространенных операционных систем в мире. Ее история насчитывает несколько десятилетий, и вот ключевые моменты ее развития:

- Windows 1.0 (1985): Первая версия Windows была выпущена в 1985 году. Она предоставляла графический интерфейс пользователя (GUI) для MS-DOS и включала приложения, такие как калькулятор и блокнот.

- Windows 3.0 (1990): Windows 3.0 внесла множество улучшений, включая поддержку цветовых экранов и возможность мультитаскинга. Она стала широко распространенной и популярной версией.

- Windows 95 (1995): Windows 95 была революционным выпуском, введя значительные изменения в интерфейс пользователя и поддерживая 32-битные приложения. Она также внесла стартовую кнопку "Пуск".

- Windows XP (2001): Windows XP была одной из самых долгожданных версий Windows и предоставила стабильность и производительность. Она была широко использована и послужила основой для многих приложений.

- Windows 7 (2009): Windows 7 улучшила интерфейс пользователя и внесла улучшения в производительность. Эта версия также осталась популярной на протяжении долгого времени.

- Windows 8 и 8.1 (2012): Эти версии внесли радикальные изменения в интерфейс, с учетом сенсорных устройств. Однако они также были критикованы за отсутствие классической кнопки "Пуск".

- Windows 10 (2015): Windows 10 была представлена как универсальная операционная система для ПК, планшетов и смартфонов. Она восстановила кнопку "Пуск" и стала одной из самых актуальных версий.

Windows имеет множество версий, включая Home, Pro, Enterprise и другие, предназначенные для разных категорий пользователей и компаний.

Windows обладает рядом значительных преимуществ и достоинств:

- Популярность и совместимость: Windows является самой популярной операционной системой, что обеспечивает широкую совместимость с программным обеспечением и аппаратным обеспечением.

- Игровая платформа: Windows является ведущей платформой для игр и обеспечивает доступ к широкому спектру видеоигр.

- Простота использования: Интерфейс Windows обычно считается более интуитивным и легким для освоения, особенно для новых пользователей.

- Множество функций: Windows предоставляет множество функций, включая встроенные инструменты для работы с офисными приложениями, веб-браузерами и мультимедиа

- Обновления и поддержка: Microsoft регулярно выпускает обновления и обеспечивает техническую поддержку для Windows, обеспечивая безопасность и стабильность системы.

- Широкий спектр аппаратных устройств: Windows поддерживает множество устройств, включая настольные ПК, ноутбуки, планшеты и смартфоны.

* 1. **Анализ программного обеспечения для написания программы и обоснование выбора платформы**

Для разработки виджета мониторинга состояния аппаратного обеспечения компьютера и прогноза времени его работы были рассмотрены несколько IDE:

- PyCharm

- Visual Studio

Был выбран именно PyCharm как основная среда разработки для языка Python. Эта мощная IDE предоставляет обширный набор инструментов для работы с Python, включая отладчик, поддержку виртуальной среды. Также PyCharm очень прост в использовании, не вызывает трудностей и имеет возможность «помогать» программисту в разработке кода, показывая специальные подсказки для упрощения работы.

Курсовой проект разрабатывается с использованием языка программирования Python и интегрированной среды разработки PyCharm также и последующим причинам. Во-первых, Python был выбран из-за существующего опыта работы с ним и предпочтения этого языка программирования. Во-вторых, PyCharm была выбрана как среда разработки из-за ее удобства и функциональности, а также из-за предпочтения этой среды разработки.

Этот выбор основывается на комфорте и знании инструментов, что способствует более эффективной разработке курсового проекта.

**CПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

[1] Архитектура вычислительных систем [Электронный ресурс]: учебное пособие – Эл. изд. - Электрон. текстовые дан. (1 файл pdf: 77 с.). - Грейбо С.В., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. 2019. – Режим доступа: <http://scipro.ru/conf/computerarchitecture.pdf>.

[2] Таненбаум, Э. Архитектура компьютера – 6-е изд. / Э. Таненбаум. СПб. : Питер, 2013. – 816 с.

[3] Петцольд, Ч. Код. Тайный язык информатики / Ч. Петцольд. – М. : Манн, Иванов и Фербер, 2021. – 448 с.

[4] Харрис, Д. М. Цифровая схемотехника и архитектура компьютера – 2-е изд. / Д. М. Харрис, С. Л. Харрис. – NY : Elsevier Inc, 2013. – 1662 c.

[5] Шитов, В. Н. Windows 10 : самый простой и понятный самоучитель / В. Шитов. – М. : Эксмо, 2023. – 464 с.

[6] Русинович, М. Внутреннее устройство Windows – 7-е изд. / М. Русинович [и др.]. – СПб. : Питер, 2018. – 944 с.

[7] Попов, А. Администрирование Windows с помощью WMI и WMIC. / А. В. Попов, Е. А. Шикин. СПб. : БХВ-Петербург, 2004. – 752 с.

[8] [Электронный ресурс]. – Режим доступа:  [https://coremission.net/gamedev/kak-rabotaet-gpu – Дата доступа 30.09.2023](https://learn.microsoft.com/ru-ru/windows/win32/api%20–%20Дата%20доступа%2020.09.2023)

[9] [Электронный ресурс]. – Режим доступа: https://www.jetbrains.com/help/pycharm/quick-start-guide - Дата доступа: 02.09.2023

[10] [Электронный ресурс]. – Режим доступа: https://internet-lab.ru/windows\_os - Дата доступа: 02.09.2023

# **ПРИЛОЖЕНИЕ А**

**(обязательное)**

**Исходный код**