Министерство образования Калининградской области

государственное бюджетное учреждение Калининградской области

профессиональная образовательная организация

«Колледж информационных технологий и строительства»

(ГБУ КО ПОО «КИТиС»)

Отчет по учебной практике

УП.04 Сопровождение и обслуживание программного обеспечения компьютерных систем

по ПМ.04 Сопровождение и обслуживание программного обеспечения компьютерных систем

Специальность 09.02.07 «Информационные системы и программирование»

Сроки прохождения практики:

с «13» февраля 2023 г. по «4» марта 2023 г.

Место практики ГБУ КО ПОО «КИТиС»

|  |  |
| --- | --- |
| Выполнил: | студент 2 курса,  группы ИСп 21-2К  Толстов Владислав Андреевич  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |
| Проверила: | Большакова-Стрекалова Анна Викторовна  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) |

Калининград, 2023

Содержание

1 Обеспечение внедрения и поддержки программного обеспечения компьютерных систем

1.1 Инструктаж по технике безопасности

1.2 Описание выбранного аппаратного и программного обеспечения

1.3 Методы внедрения и анализа функционирования программного обеспечения

1.4 Сценарий и техническое задание на внедрение ПО

1.5 Загрузка и установка программного обеспечения

1.6 Настройка отдельных компонентов программного обеспечения компьютерных систем и конфигурирование

2 Проблемы совместимости и методы их устранения

2.1 Тестирование на совместимость в безопасном режиме

2.2 Инструменты повышения производительности программного обеспечения

2.3 Средства диагностики оборудования

2.4 Измерение и анализ эксплуатационных характеристик качества программного обеспечения

2.5 Оптимизация и модификация ПО

2.6 Разработка руководства оператора

3 Этапы сопровождения и обслуживания ПО

3.1 Разработка технического задания на сопровождение

3.2 Обеспечение качества компьютерных систем в процессе эксплуатации

3.3 Разработка сопровождающей документации

4 Методы и средства защиты компьютерных систем

4.1 Тестирование на ошибки ввода/вывода данных

4.2 Тестирование интерфейса

4.3 Тестирование контента

4.4 Анализ рисков. Выявление первичных и вторичных ошибок

4.5 Обнаружение вируса и устранение последствий его влияния

4.6 Тестирование защиты программного обеспечения

4.7 Работа с реестром. Работа с программой восстановления файлов и очистки дисков

5. Индивидуальное задание [5]

5.1 Практические работы

5.2 \*(Теоретический вопрос)

5.3 \*(Самостоятельная работа)

Введение

Учебная практика по ПМ 04 «Сопровождение и обслуживание программного обеспечения компьютерных систем»

Цель: выполнить задания, научиться внедрять продукт

Shotcut – это свободный видеоредактор для FreeBSD, Linux, macOS и Windows. Поддерживает множество видео, аудио и графических форматов. Использует шкалу времени для нелинейного редактирования из нескольких дорожек, которые могут состоять из файлов различных форматов.

Основные возможности Shotcut:

С помощью **Shotcut** можно создавать коллажи из видеороликов, редактировать их с помощью широкого набора настраиваемых фильтров (аудио и видео), корректировать качество видео и аудиорядов, изменять громкость, выбирать желаемую частоту дискретизации.

Достоинства Shotcut:

* Большой функционал
* Кроссплатформенность
* Полностью бесплатная программа
* Удобный интерфейс

Недостатки Shotcut:

* Неудобная реализация переходов между видеорядами
* Программа может вылететь в неожиданный момент
  1. Инструктаж по технике безопасности

1. Необходимо постоянно следить на своем рабочем месте за исправным состоянием электропроводки, выключателей, штепсельных розеток, при помощи которых оборудование включается в сеть, и заземления. При обнаружении неисправности немедленно обесточить электрооборудование, оповестить администрацию. Продолжение работы возможно только после устранения неисправности.

2. Во избежание повреждения изоляции проводов и возникновения коротких замыканий не разрешается:

а) вешать что-либо на провода;

б) закрашивать и белить шнуры и провода;

в) закладывать провода и шнуры за газовые и водопроводные трубы, за батареи отопительной системы;

г) выдергивать штепсельную вилку из розетки за шнур, усилие должно быть приложено к корпусу вилки.

3. Для исключения поражения электрическим током запрещается:

а) часто включать и выключать компьютер без необходимости;

б) прикасаться к экрану и к тыльной стороне блоков компьютера;

в) работать на средствах вычислительной техники и периферийном оборудовании мокрыми руками;

г) работать на средствах вычислительной техники и периферийном оборудовании, имеющих нарушения целостности корпуса, нарушения изоляции проводов, неисправную индикацию включения питания, с признаками электрического напряжения на корпусе

д) класть на средства вычислительной техники и периферийном оборудовании посторонние предметы.

4. Запрещается под напряжением очищать от пыли и загрязнения электрооборудование.

5. Запрещается проверять работоспособность электрооборудования в неприспособленных для эксплуатации помещениях с токопроводящими полами, сырых, не позволяющих заземлить доступные металлические части.

6. Ремонт электроаппаратуры производится только специалистами-техниками с соблюдением необходимых технических требований.

7. Недопустимо под напряжением проводить ремонт средств вычислительной техники и периферийного оборудования.

8. Во избежание поражения электрическим током, при пользовании электроприборами нельзя касаться одновременно каких-либо трубопроводов, батарей отопления, металлических конструкций, соединенных с землей.

9. При пользовании электроэнергией в сырых помещениях соблюдать особую осторожность.

10. При обнаружении оборвавшегося провода необходимо немедленно сообщить об этом администрации, принять меры по исключению контакта с ним людей. Прикосновение к проводу опасно для жизни.

11. При выполнении работ нужно соблюдать следующие правила:

* запрещается класть на корпус и дисплей компьютера посторонние предметы, прикасаться к элементам аппаратуры мокрыми руками, производить чистку корпуса оборудования, находящегося под напряжением, располагать технику близко к жилищно-коммунальным инженерным системам;
* в случае обнаружения неисправности компьютера немедленно прекратить работу и сообщить об этом непосредственному руководителю;
* эксплуатировать компьютер только с соблюдением инструкции, установленной производителем;
* избегать частого и необоснованного включения и выключения компьютера во время работы.

1.2 Описание выбранного аппаратного и программного обеспечения

Shotcut – это бесплатный кроссплатформенный видеоредактор с открытым исходным кодом для Windows, Mac и Linux. Основные функции включают поддержку широкого спектра форматов; импорт не требуется, что означает встроенное редактирование временной шкалы; Blackmagic Design поддерживает мониторинг ввода и предварительного просмотра; и поддерживает разрешение до 4k.

Язык интерфейса: Английский, Русский.  
Системные требования (минимальные): 32 или 64-битный двухъядерный процессор c тактовой частотой 1,2 ГГц, 4 ГБ ОЗУ, видеокарта с поддержкой OpenGL 2.0  
Поддержка операционных систем: GNU/Linux, Mac OS и Microsoft Windows.

Достоинства:

* Возможность вывода предпросмотра проекта на 2 монитора;
* Поддержка 4K разрешения;
* Большая библиотека инструкций и руководств по работе с программой;
* Поддержка большинства современных медиаконтейнеров и кодеков;
* Поддержка Windows DirectShow и внешних устройств захвата видео;
* Возможность работы с IP и веб-камерами.

Недостатки:

* Проблемы со стабильностью в Windows версии;
* Маленький выбор фильтров, анимаций и эффектов.

1.3 Методы внедрения и анализа функционирования программного обеспечения

Основные этапы внедрения ПО:

1. Обследование

2. Разработка технического задания

3. Настройка системы

4. Тестирование системы

5. Опытная эксплуатация

6. Промышленная эксплуатация

Первый этап проекта – диагностика предприятия или его обследование. Под обследованием подразумевается диагностика на предприятии всех бизнес-процессов, которые будут охватывать будущая система.

Второй этап – разработка технического задания. Техническое задание включает в себя описание всех справочников системы, всех алгоритмов расчета, отчетных форм, пользователей и описания разграничения прав доступа пользователей.

Третий этап – настройка системы. Настройка системы включает в себя формирование в программе всех справочников системы, настройка всех алгоритмов расчета, форм ввода и отчетных форм, ввод пользователей системы и настройка прав доступа.

Четвертый этап – тестирование ПО (системы). Тестирование системы включает в себя подготовку демонстративного примера, внесение тестовых данных, проверку алгоритмов расчета и исправление ошибок.

Пятый этап – опытная эксплуатация системы, которая включает в себя работу с реальными данными, но при этом параллельно используется прежняя старая система

Шестой этап – промышленная эксплуатация системы, которая подразумевает переход предприятия на новый программный продукт и отказ от всех альтернативных способов работы за рамками данной системы.

1.4 Сценарий и техническое задание на внедрение ПО

Этап 1. Обследование компании

Должность: Программист

Количество: 3 месяца

Сроки: с 01.09.2022 до 01.12.2022

Стоимость: 100000₽

Этап 2. Составление контракта, формирование группы внедрения

Состав – 10 сотрудников

Этап 3. Инсталляция и наладка ПО.

 Компанией-исполнителем формируется группа внедрения программного обеспечения и назначаются ответственные.

Состав группы внедрения: 10 человек

Сроки: 20 дней

Стоимость: 25000₽

Этап 4. Эксплуатация и сопровождение ПО.

* Установка ПО.
* Создание БД.
* Перенос данных.
* Интеграция с имеющимися системами.
* Конечное тестирование системы.
* Окончательная настройка.

Техническое задание

Введение

Наименование ПО – Shotcut

Разработчик – Толстов Владислав Андреевич

Назначение

1. Разработка ПО.
2. Обновление исходников, исправление проблем и т.д.

Требования к приложению

Системные требования Shotcut

Операционная система – Windows 7/8/8.1/10.

Жесткий диск – Требуется 300 МБ свободного места.

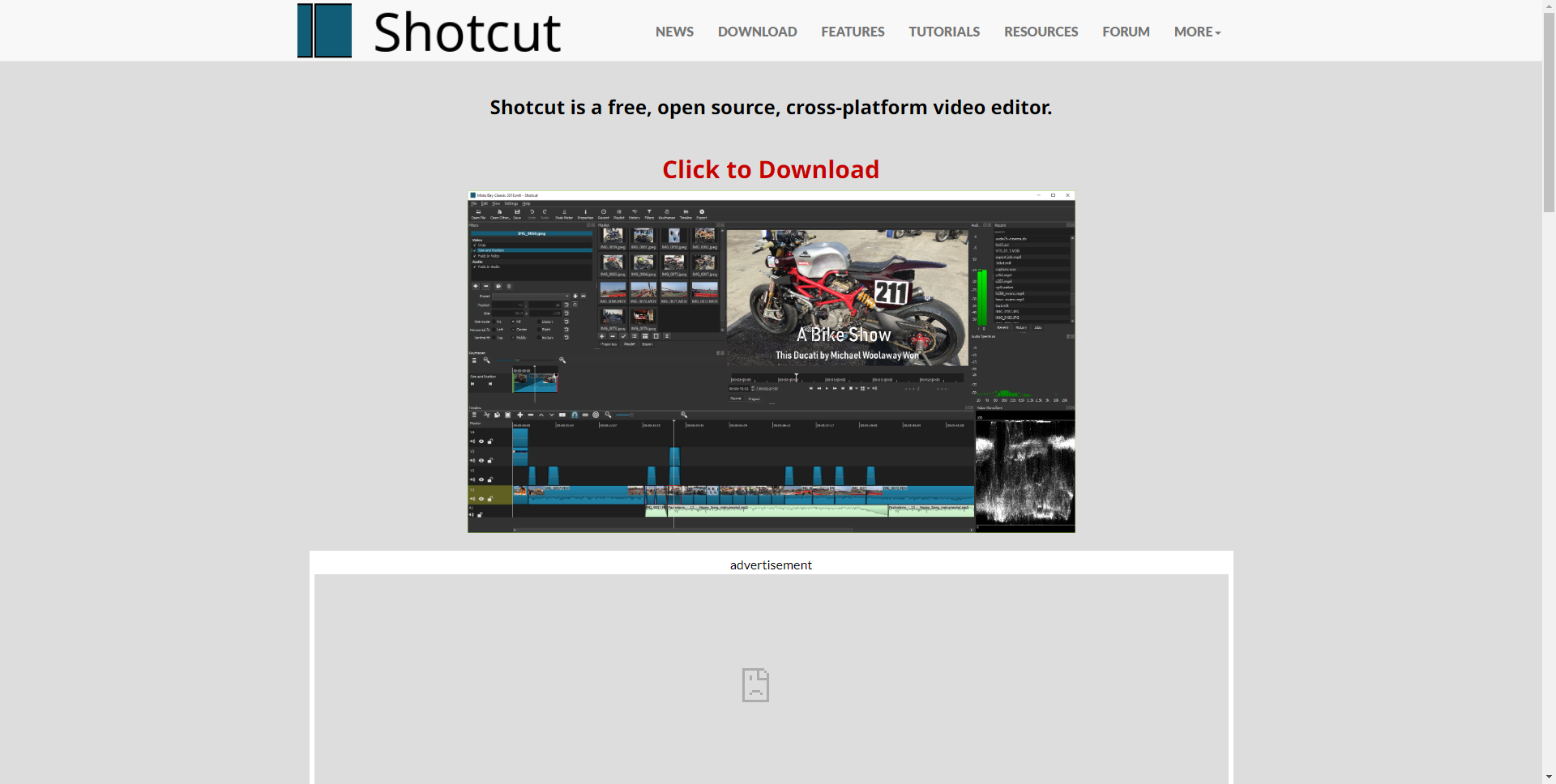
Память (ОЗУ): Требуется 1 ГБ оперативной памяти.

Процессор – Двухъядерный процессор Intel или более поздняя версия.

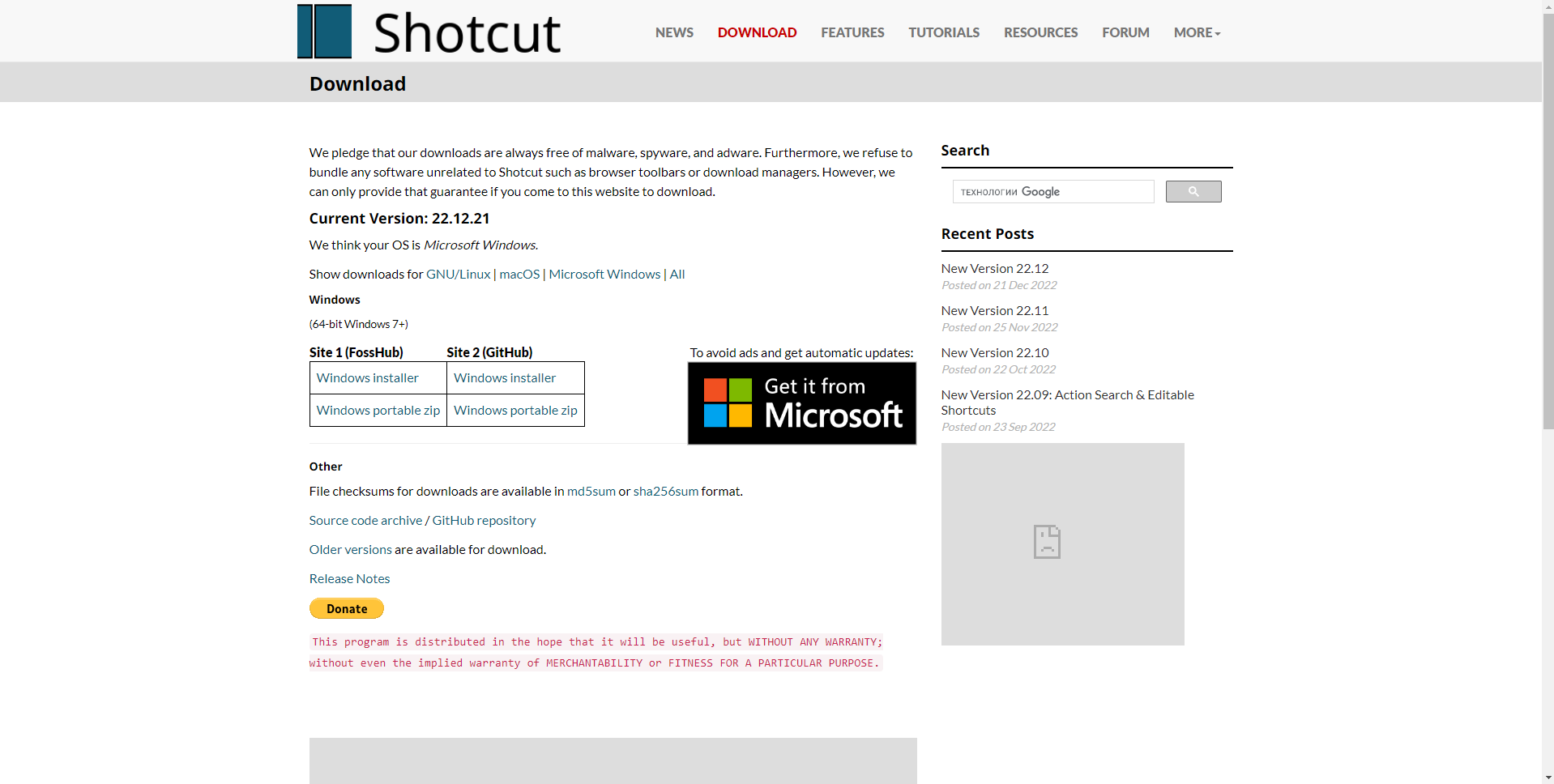
1.5 Загрузка и установка программного обеспечения

Установка Shotcut

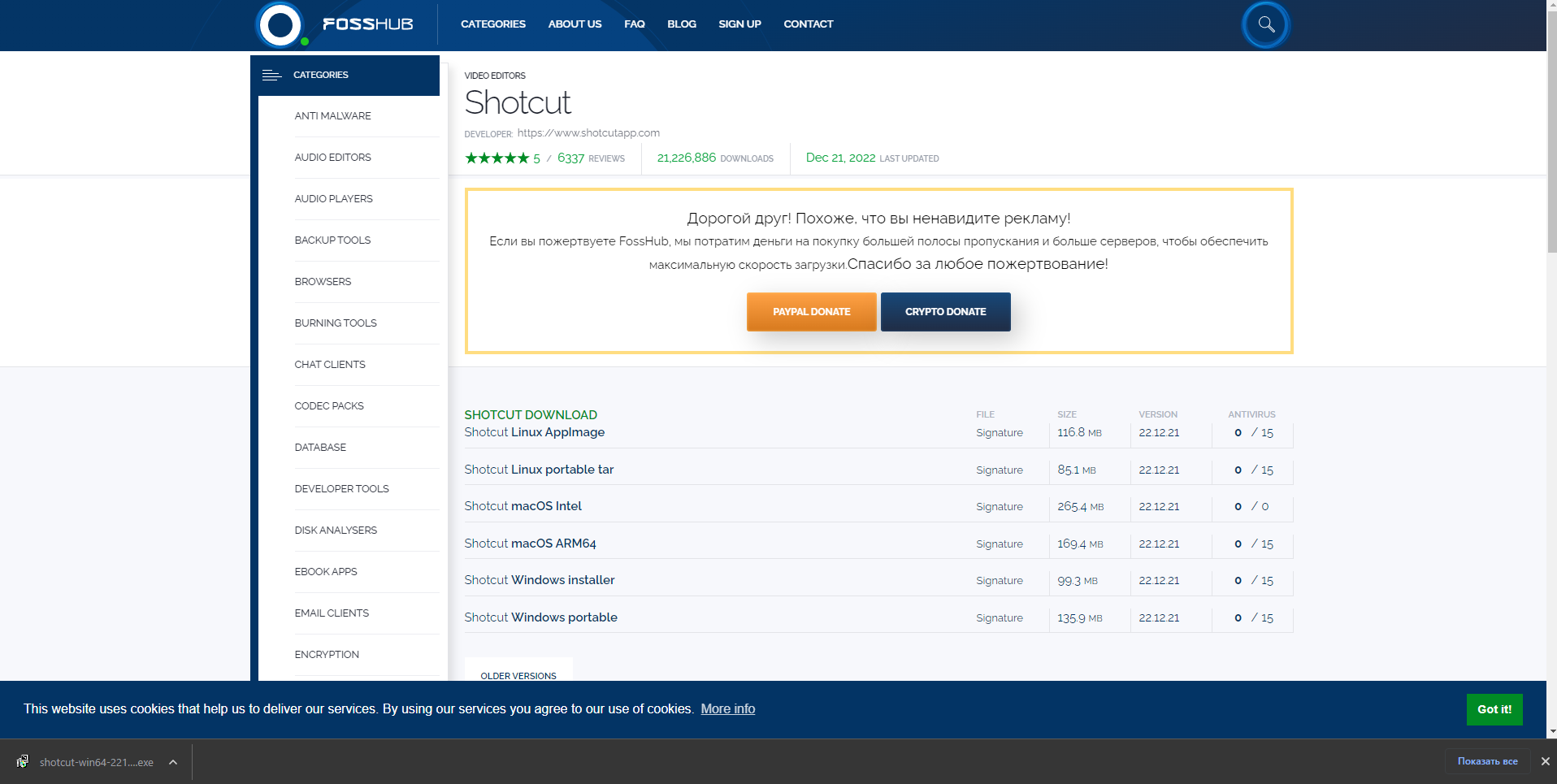
Шаг 1: перейти на сайт [shotcut.org](https://shotcut.org/)



Шаг 2: нажать CLICK TO DOWNLOAD или нажать DOWNLOAD сверху сайта

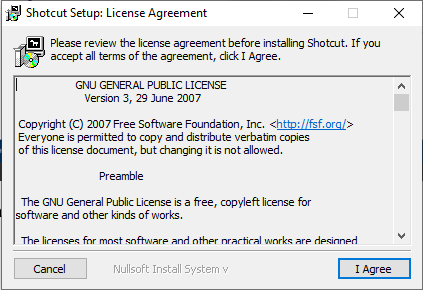


Шаг 3: выбрать желаемый installer и скачать

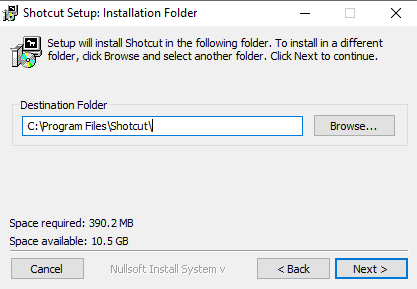


Шаг 4: запустить скачанный файл

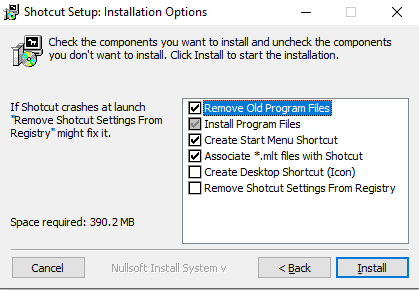
Шаг 5: прочитать лицензионное соглашение и нажать кнопку «I Agree»



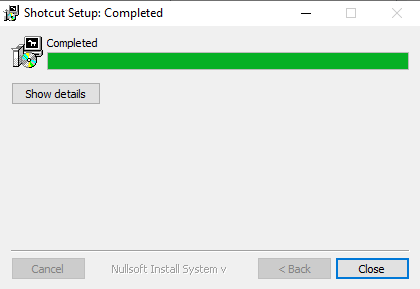
Шаг 6: указать путь установки файлов или оставить по умолчанию и нажать «Next»



Шаг 7: выбрать компоненты, которые нужно установить и нажать «Install»



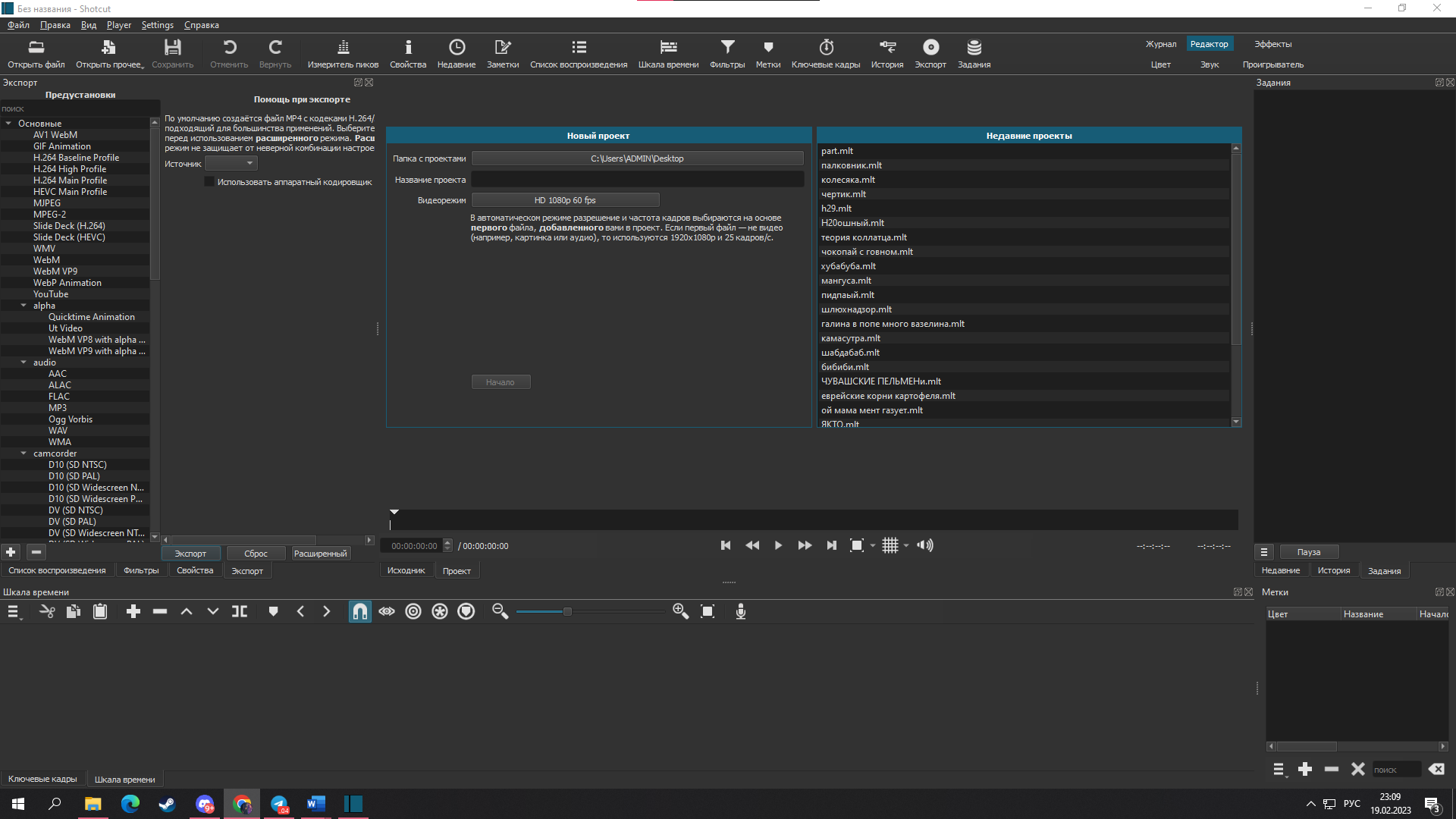
Шаг 8: после завершения установки нажать «Close»



Программа Shotcut была успешна установлена. Теперь ей можно пользоваться.

1.6 Настройка отдельных компонентов программного обеспечения компьютерных систем и конфигурирование

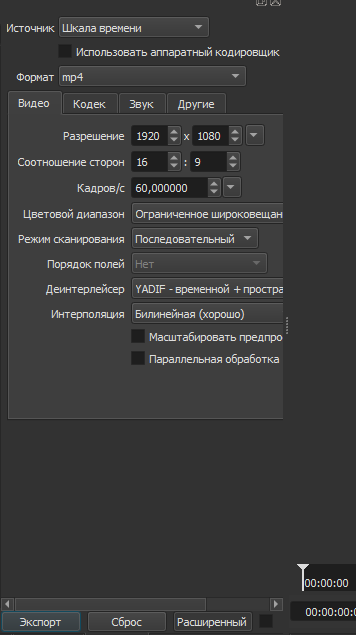
При первом запуске программы сильно заострять внимание на настройке отдельных компонентов и конфигураций не требуется, так как таких настроек практически нет и их можно изменять в любое время.



Чтобы начать работу, необходимо ввести название проекта, выбрать путь сохранения и указать нужный видеорежим.

После создания нового проекта можно приступать к самой работе без каких-либо настроек, кроме визуальных:

Вся основная рабочая зона программы разделяется на функциональные зоны:

1. По левую часть ряда расположены окно предпросмотра и фильтры для аудио и видео. Чтобы их загрузить, можно воспользоваться соответствующими клавишами. Правым кликом вызываются доступные параметры для корректирования фильтра.
2. В правой части окна инструментов представлена область истории действий, чтобы редактировать в нужный момент процесс настройки ряда, а также задания и недавние.
3. По центру основного окна ПО отображается видео с линией прокрутки и кнопками управления для воспроизведения.

Все эти окна можно регулировать по ширине для удобства использования.

При экспорте проекта стоит выбрать вариант «Расширенный» и настроить параметры так, как нужно

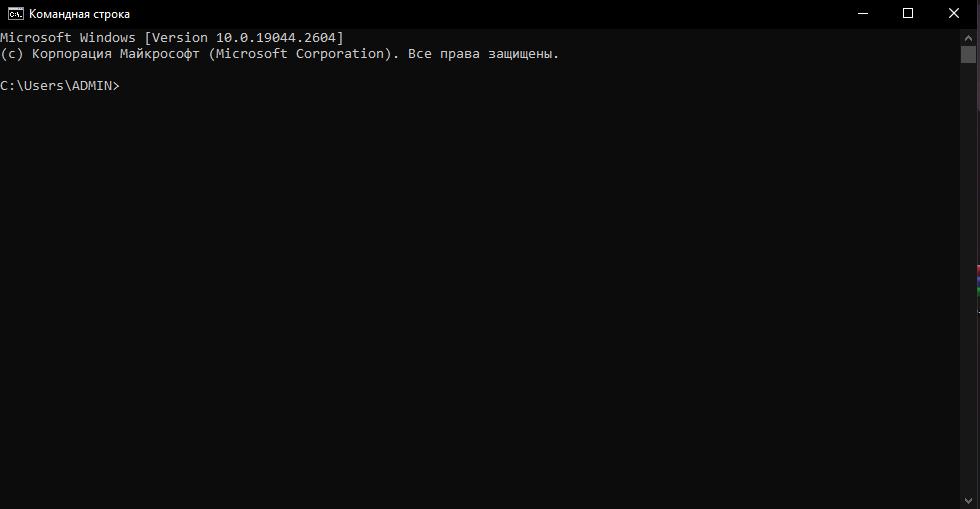
Можно оставить здесь настройки по умолчанию и нажать кнопку «Экспорт», затем выбрать путь сохранения файла и начнётся экспортирование.

* 1. Тестирование на совместимость в безопасном режим

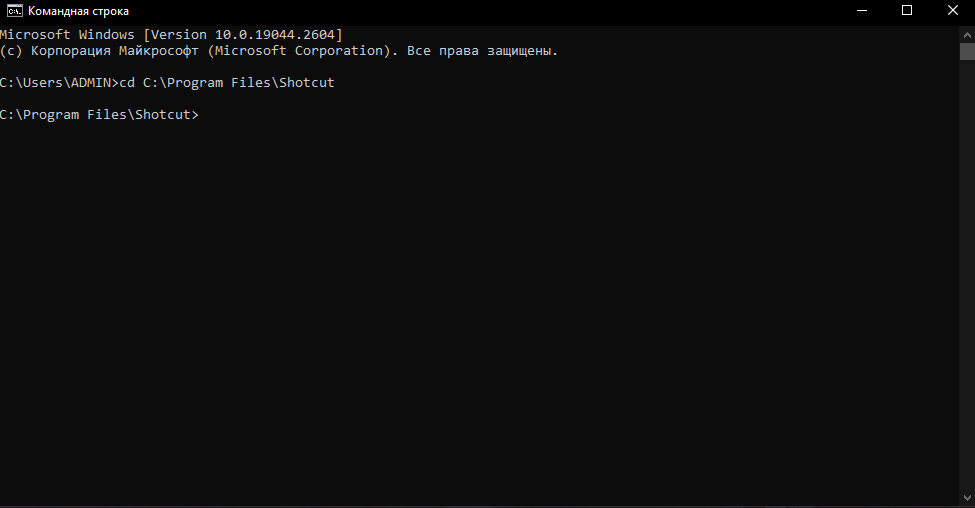
Тестирование Shotcut на совместимость в безопасном режиме может помочь выявить проблемы, связанные с совместимостью программы с другими приложениями и настройками операционной системы.

Для запуска Shotcut в безопасном режиме:

1. Закрыть все запущенные экземпляры Shotcut.
2. Запустить командную строку Windows, нажав клавишу Win + R и введя cmd.



1. В командной строке выполнить команду cd C:\Program Files\Shotcut, если Shotcut установлен в папке Program Files, или заменить путь к установленной директории Shotcut.



1. Ввести команду shotcut.exe --safe-mode и нажать Enter.
2. Shotcut запустится в безопасном режиме.

В безопасном режиме Shotcut запускается без использования настроек, плагинов и других расширений, которые могут вызывать проблемы совместимости. Это может помочь выявить проблемы, которые могут быть связаны с настройками или расширениями, а не с самой программой.

Если Shotcut работает без проблем в безопасном режиме, это может указывать на проблемы совместимости с другими приложениями или настройками операционной системы. В этом случае, вы можете попробовать изменить настройки или удалить конфликтующее приложение.

В целом, тестирование Shotcut на совместимость в безопасном режиме может помочь выявить проблемы совместимости, которые могут быть связаны с настройками или расширениями, а не с самой программой, и помочь устранить эти проблемы.

* 1. Инструменты повышения производительности программного обеспечения

Shotcut – это мощный видеоредактор, который может потреблять значительные ресурсы компьютера. Однако, существуют инструменты, которые могут помочь повысить производительность программы, ускорить работу и уменьшить нагрузку на компьютер. Ниже перечислены некоторые из этих инструментов:

1. Уменьшение размера кэша: Shotcut хранит кэш файлов, которые помогают ускорить обработку видео и аудио. Однако, если размер кэша установлен слишком большим, это может привести к снижению производительности. Уменьшение размера кэша может помочь ускорить работу программы.
2. Использование производительного компьютера: Shotcut требует значительных ресурсов компьютера, поэтому использование более мощного компьютера, особенно с более быстрым процессором и графической картой, может значительно ускорить работу программы.
3. Использование промежуточных файлов: Работа с большими файлами видео и аудио может быть медленной и нагружать компьютер. Использование промежуточных файлов может помочь ускорить работу программы, особенно при работе с большими файлами.
4. Использование правильных форматов файлов: Shotcut поддерживает множество форматов файлов, но некоторые из них более нагружают процессор, чем другие. Использование форматов файлов, которые меньше всего нагружают процессор, таких как H.264 или ProRes, может помочь ускорить работу программы.
5. Уменьшение числа потоков обработки: Shotcut поддерживает множество потоков обработки, но установка слишком большого числа потоков может привести к снижению производительности. Уменьшение числа потоков обработки может помочь ускорить работу программы.
6. Использование GPU ускорения: Shotcut поддерживает использование графических процессоров для ускорения обработки видео. Включение этой опции может ускорить работу программы на компьютерах с поддержкой GPU.

В целом, эти инструменты могут помочь улучшить производительность Shotcut и ускорить работу программы. Однако, для каждого конкретного случая могут потребоваться индивидуальные настройки и оптимизации.

* 1. Средства диагностики оборудования

## Для диагностики оборудования можно скачать специальную утилиту CPU-Z. Программа показывает информацию о процессоре, материнской плате, системной памяти. Использовать эту программу можно на Windows, также есть версия для Android.

## Также специально для диагностики видеокарты существует GPU-Z

* 1. Измерение и анализ эксплуатационных характеристик качества программного обеспечения

Оценка качества пользователем производится на этапах внедрения, сопровождения и эксплуатации ПС.

1. Производительность: Shotcut довольно быстро загружается и обрабатывает файлы, однако, скорость обработки зависит от размера и качества видеофайла, а также от производительности компьютера.
2. Интерфейс: Интерфейс программы Shotcut интуитивно понятный и легко настраиваемый, однако, он может показаться сложным для начинающих пользователей. Есть достаточное количество инструментов и функций, доступных через меню и панели инструментов.
3. Функциональность: Shotcut предоставляет множество инструментов для редактирования видео, таких как обрезка, изменение размера, наложение текста и звуковых эффектов, добавление фильтров и другие. Кроме того, программа может работать с широким спектром форматов файлов и предоставляет возможность создавать проекты различных разрешений.
4. Совместимость: Shotcut имеет хорошую совместимость с другими приложениями и форматами файлов, включая форматы, такие как MP4, MOV, AVI, WAV, и другие.
5. Стабильность: Shotcut, как правило, довольно стабильная программа, однако, иногда могут возникать проблемы с зависанием или ошибками, особенно если программа используется для обработки больших и сложных видеофайлов.

В целом, Shotcut – это хорошая программа для редактирования видео, которая имеет множество функций и хорошую совместимость с различными форматами файлов. Стоит учитывать, что характеристики качества могут зависеть от мощности компьютера.

* 1. Оптимизация и модификация ПО

Shotcut является открытым исходным кодом программы, что означает, что любой может скачать, изменять и распространять ее исходный код. Это также означает, что программа может быть оптимизирована и модифицирована в соответствии с потребностями конкретного пользователя или сообщества.

1. Оптимизация производительности: для улучшения производительности Shotcut, можно использовать следующие советы:

* Использование более мощного компьютера, особенно с более быстрым процессором и графической картой;
* Использование оптимизированных настроек программы, таких как уменьшение размера кэша и использование меньшего числа потоков обработки;
* Использование форматов файлов, которые меньше всего нагружают процессор, таких как H.264 или ProRes.

1. Модификация интерфейса: Интерфейс Shotcut может быть модифицирован, чтобы удовлетворить индивидуальные потребности пользователя или сообщества. Например, можно изменить цветовую схему, настроить расположение панелей инструментов, изменить шрифты и т.д.
2. Модификация функциональности: Shotcut может быть модифицирован, чтобы добавить новые функции, убрать ненужные, или изменить существующие. Это может быть полезно для пользователей, которые нуждаются в функциях, которые не входят в стандартный набор инструментов программы.
3. Разработка плагинов: Shotcut поддерживает разработку плагинов, что позволяет расширять функциональность программы. Плагины могут добавлять новые эффекты, фильтры, инструменты и т.д. и помогают улучшить возможности программы.

Общая оптимизация и модификация программы Shotcut зависит от конкретных потребностей пользователя или сообщества. Но благодаря тому, что Shotcut является открытым исходным кодом, пользователи могут внести изменения в программу, чтобы она лучше соответствовала их потребностям.

* 1. Разработка руководства оператора

Shotcut является свободно распространяемым кросс-платформенным видеоредактором, который обладает широким набором функций и возможностей для редактирования видео. Целью данного руководства является предоставление оператору всей необходимой информации о программе Shotcut и ее возможностях, а также предоставление инструкций по использованию всех функций программы.

Интерфейс программы

Интерфейс программы Shotcut состоит из нескольких основных элементов, включая панель инструментов, окна настроек, окно монтажа и другие. Панель инструментов содержит инструменты, необходимые для редактирования видео, такие как инструменты разделения, соединения и обрезки клипов, наложения фильтров и эффектов, а также инструменты для добавления титров и переходов.

Импорт медиафайлов

Чтобы импортировать медиафайлы в программу, необходимо выбрать опцию "Открыть файл" в меню "Файл" или перетащить файлы непосредственно в окно монтажа. Shotcut поддерживает множество форматов файлов, включая AVI, MOV, MP4 и другие.

Монтаж видео

Для монтажа видео в Shotcut доступны различные функции, включая разделение и соединение клипов, изменение скорости и размера клипов, наложение фильтров и эффектов на видео и другие. Инструменты монтажа доступны на панели инструментов и могут быть использованы с помощью мыши.

Создание титров и переходов

Shotcut позволяет создавать титры и переходы в видео, чтобы сделать его более профессиональным и привлекательным. Для создания титров и переходов необходимо выбрать соответствующий инструмент на панели инструментов и следовать инструкциям на экране.

Экспорт видео

Shotcut позволяет экспортировать видео в различные форматы, включая MP4, AVI, MOV и другие. Чтобы экспортировать видео, необходимо выбрать опцию "Экспортировать" в меню "Файл" и выбрать желаемый формат экспорта, разрешение и другие параметры.

Использование дополнительных инструментов

Shotcut также содержит дополнительные инструменты, которые могут быть использованы для редактирования

3.1 Разработка технического задания на сопровождение

Техническое задание на сопровождение Shotcut

1. Введение
   * Краткое описание продукта и его особенностей: Shotcut – это свободно распространяемый кроссплатформенный видеоредактор, который предоставляет пользователям возможность создавать и редактировать видео в различных форматах. Он поддерживает множество функций, таких как обрезка, наложение фильтров, корректировка яркости и контрастности, добавление звуковых эффектов и многие другие.
   * Обоснование необходимости сопровождения: Shotcut является важным инструментом для многих пользователей, которые используют его для создания и редактирования видео. Разработчики Shotcut продолжают выпускать новые версии продукта и исправлять ошибки, но для обеспечения дальнейшей стабильной работы и улучшения продукта необходимо организовать сопровождение.
2. Общие требования к сопровождению
   * Описание задач, которые должны быть выполнены в рамках сопровождения:
     + Исправление ошибок, выявленных в процессе эксплуатации продукта.
     + Внесение изменений в продукт по запросу пользователей или в связи с изменениями в окружающей среде.
     + Разработка и релиз новых версий продукта, включающих улучшения и новые функции.
     + Поддержка и обновление документации, связанной с продуктом.
     + Поддержка пользователей и обучение их новым функциям и возможностям продукта.
   * Определение ответственности разработчика и заказчика:
     + Разработчик Shotcut несет ответственность за исправление ошибок, разработку новых версий продукта и поддержку документации.
     + Заказчик (в данном случае, организация, которая финансирует разработку) несет ответственность за обеспечение инфраструктуры и окружения для сопровождения, обучение пользователей и поддержку пользователей.
3. Технические требования
   * Описание технических характеристик продукта:
     + Shotcut должен работать на различных операционных системах (Windows, macOS, Linux) и поддерживать различные форматы файлов.

Продукт должен обладать высокой производительностью и не должен тормозить при работе с большими файлами или сложными проектами. - Продукт должен иметь удобный и интуитивно понятный интерфейс для пользователей разного уровня подготовки. - Продукт должен поддерживать современные технологии и стандарты, такие как GPU-ускорение и 4K-видео.

* Требования к обеспечению безопасности и надежности:
  + Продукт должен обеспечивать защиту от возможных угроз безопасности, таких как вирусы или злоумышленники.
  + Разработчики должны регулярно выпускать обновления, исправляющие возможные уязвимости и проблемы безопасности.
  + Продукт должен иметь систему резервного копирования данных и возможность восстановления после сбоев и ошибок.
* Требования к процессу тестирования:
  + Разработчики должны регулярно проводить тестирование новых версий продукта перед их выпуском.
  + Тестирование должно включать проверку основных функций и возможностей продукта, а также поиск и устранение возможных ошибок и проблем.
  + Тестирование должно проводиться на различных платформах и в различных условиях эксплуатации.

1. План сопровождения
   * Описание плана релизов новых версий:
     + Разработчики должны выпускать новые версии продукта регулярно, не реже чем раз в полгода.
     + Каждая новая версия должна содержать улучшения и новые функции, а также исправления ошибок.
     + Разработчики должны предоставлять пользователям возможность бесплатного обновления до новых версий.
   * Описание процесса обработки запросов от пользователей:
     + Разработчики должны создать систему для приема и обработки запросов от пользователей, связанных с продуктом.
     + Каждый запрос должен быть проанализирован и решен в соответствии с приоритетами и сроками выполнения.
   * Описание процесса технической поддержки:
     + Разработчики должны предоставлять пользователей поддержку в случае возникновения проблем при работе с продуктом.
     + Поддержка может быть оказана посредством электронной почты, форумов или чата.

* Разработчики должны отвечать на запросы пользователей в течение определенного времени, например, в течение 24 часов.
* Описание процесса обновления и установки продукта:
  + Разработчики должны обеспечить удобный и безопасный процесс установки и обновления продукта.
  + Пользователи должны иметь возможность получать обновления автоматически или скачивать и устанавливать их вручную.
  + Пользователи должны получать инструкции по установке и обновлению продукта.
* Описание процесса мониторинга и анализа работы продукта:
  + Разработчики должны проводить мониторинг работы продукта с целью выявления проблем и улучшения его работы.
  + Мониторинг может включать сбор информации о частоте использования функций продукта, количестве ошибок и сбоев, а также о производительности.
  + Разработчики должны анализировать полученную информацию и принимать меры для улучшения продукта.

1. Требования к документации
   * Разработчики должны предоставить полную и понятную документацию по продукту, включая:
     + Руководство пользователя, описывающее основные функции и возможности продукта.
     + Руководство администратора, описывающее процесс установки и настройки продукта.
     + Руководство разработчика, описывающее API и возможности расширения продукта.
     + Справочную документацию по функциям и настройкам продукта.
   * Документация должна быть доступна на русском языке.
2. Требования к персоналу
   * Разработчики, занимающиеся сопровождением продукта, должны иметь следующие квалификации:
     + Опыт работы с программированием на языке C++ и Qt Framework.
     + Знание основных алгоритмов и структур данных.
     + Опыт работы с системами контроля версий, такими как Git.
     + Знание основных принципов разработки ПО и процессов тестирования.
     + Умение работать в команде и общаться с пользователями.
   * Персонал, обеспечивающий техническую поддержку, должен иметь следующие квалификации:
     + Знание основных функций и возможностей продукта.
     + Опыт работы с операционными системами Windows, macOS и Linux.

* Умение общаться с пользователями и решать технические проблемы.
* Персонал должен иметь возможность обучения и повышения квалификации в соответствии с развитием продукта и изменениями в технологиях.

1. Требования к процессу управления изменениями
   * Разработчики должны обеспечить процесс управления изменениями, чтобы изменения в продукте вносились систематически и контролируемо.
   * Любые изменения в продукте должны быть документированы и проходить процесс тестирования перед внедрением.
   * Разработчики должны обеспечивать процесс отката изменений в случае неудачного внедрения.
2. Требования к обеспечению безопасности и защите информации
   * Разработчики должны обеспечивать безопасность продукта и защиту информации пользователей.
   * Продукт должен иметь механизмы защиты от несанкционированного доступа и вредоносного ПО.
   * Разработчики должны регулярно выпускать обновления, включающие исправления уязвимостей и обеспечивающие безопасность продукта.
   * Разработчики должны обеспечивать защиту конфиденциальной информации пользователей, такой как логины и пароли.

3.2 Обеспечение качества компьютерных систем в процессе эксплуатации

Для обеспечения качества компьютерных систем в процессе эксплуатации Shotcut необходимо уделить внимание следующим аспектам:

1. Тестирование продукта перед выпуском
   * Перед выпуском новой версии продукта необходимо провести тестирование, чтобы обеспечить корректную работу всех функций и убедиться в отсутствии ошибок и непредвиденных сбоев.
   * Необходимо проводить тестирование на разных конфигурациях оборудования и операционных системах.
2. Мониторинг работы продукта в процессе эксплуатации
   * Необходимо осуществлять мониторинг работы продукта в процессе эксплуатации, чтобы своевременно обнаруживать и устранять возникающие проблемы.
   * Мониторинг должен включать следующие аспекты: использование ресурсов системы (процессор, память, жесткий диск), работу сетевых подключений, логирование ошибок и сбоев.
3. Проведение регулярных обновлений и исправлений
   * Для обеспечения стабильной и безопасной работы продукта необходимо регулярно выпускать обновления, включающие исправления ошибок и уязвимостей.
   * Обновления должны быть проверены перед выпуском, чтобы убедиться в их корректности и отсутствии негативного влияния на работу продукта.
4. Обеспечение безопасности и защиты информации
   * Разработчики должны обеспечивать безопасность продукта и защиту информации пользователей.
   * Продукт должен иметь механизмы защиты от несанкционированного доступа и вредоносного ПО.
   * Разработчики должны регулярно выпускать обновления, включающие исправления уязвимостей и обеспечивающие безопасность продукта.
   * Разработчики должны обеспечивать защиту конфиденциальной информации пользователей, такой как логины и пароли.
5. Обеспечение доступности продукта
   * Разработчики должны обеспечивать доступность продукта для пользователей в любое время.
   * Необходимо обеспечить резервное копирование данных и системы, чтобы была возможность восстановления работы продукта в случае отказа оборудования или сбоев в работе.
6. Обеспечение производительности
   * Продукт должен обеспечивать высокую производительность и отзывчивость в процессе работы.
   * Разработчики должны оптимизировать работу продукта, устранять узкие места в производительности и минимизировать использование ресурсов системы.
7. Оказание технической поддержки
   * Разработчики должны оказывать техническую поддержку пользователям в случае возникновения проблем с работой продукта.
   * Техническая поддержка должна осуществляться квалифицированными специалистами, которые могут решить любые возникающие проблемы.
8. Слежение за трендами и развитием технологий
   * Разработчики должны следить за развитием технологий и новыми трендами в сфере разработки программного обеспечения.
   * Необходимо регулярно обновлять продукт и внедрять новые функции и возможности, чтобы соответствовать требованиям пользователей и рынка.
9. Разработка документации
   * Разработчики должны разрабатывать и поддерживать документацию на продукт, которая будет содержать все необходимые инструкции и руководства по использованию продукта.
   * Документация должна быть доступна для пользователей на официальном сайте и должна содержать подробное описание функций и возможностей продукта.
10. Тестирование продукта

* Разработчики должны проводить тестирование продукта на всех поддерживаемых платформах и операционных системах, чтобы убедиться в его работоспособности и отсутствии ошибок.
* Тестирование должно проводиться как перед выпуском новых версий продукта, так и в процессе эксплуатации для выявления и исправления возникающих проблем.

1. Обеспечение безопасности

* Продукт должен обеспечивать высокий уровень безопасности для пользователей.
* Разработчики должны следить за обновлениями безопасности и оперативно выпускать исправления уязвимостей в случае их обнаружения.

1. Развитие и улучшение продукта

* Разработчики должны постоянно улучшать и развивать продукт, добавляя новые функции и возможности, оптимизируя работу и обеспечивая удобство использования для пользователей.
* Необходимо регулярно проводить исследования рынка и мониторинг конкурентов, чтобы определить новые требования и возможности для улучшения продукта.

3.3 Разработка сопровождающей документации

1. Введение В данном разделе необходимо описать цель и задачи сопровождения продукта, а также предоставить общую информацию о продукте.
2. Руководство пользователя Руководство пользователя – это основной документ, который предназначен для обучения пользователей работе с продуктом. Оно должно содержать следующие разделы:

* Установка и настройка продукта;
* Описание интерфейса программы;
* Описание основных функций продукта;
* Описание процесса работы с проектами;
* Описание экспорта и импорта файлов;
* Решение основных проблем.

1. Руководство оператора Руководство оператора предназначено для обучения специалистов, ответственных за сопровождение программного продукта. В нем должны быть описаны следующие разделы:

* Описание архитектуры продукта;
* Описание внутренних механизмов работы программы;
* Описание процесса разработки и тестирования продукта;
* Описание поддержки пользователей и решения проблем.

1. Документация по API Документация по API предназначена для разработчиков, которые хотят интегрировать продукт в свои проекты. Она должна содержать следующие разделы:

* Описание интерфейсов программного продукта;
* Описание процесса взаимодействия с API.

1. Документация по обновлениям Документация по обновлениям содержит информацию о новых версиях продукта и исправлениях, включая:

* Описание новых функций и улучшений;
* Описание исправлений и уязвимостей;
* Инструкции по установке обновлений.

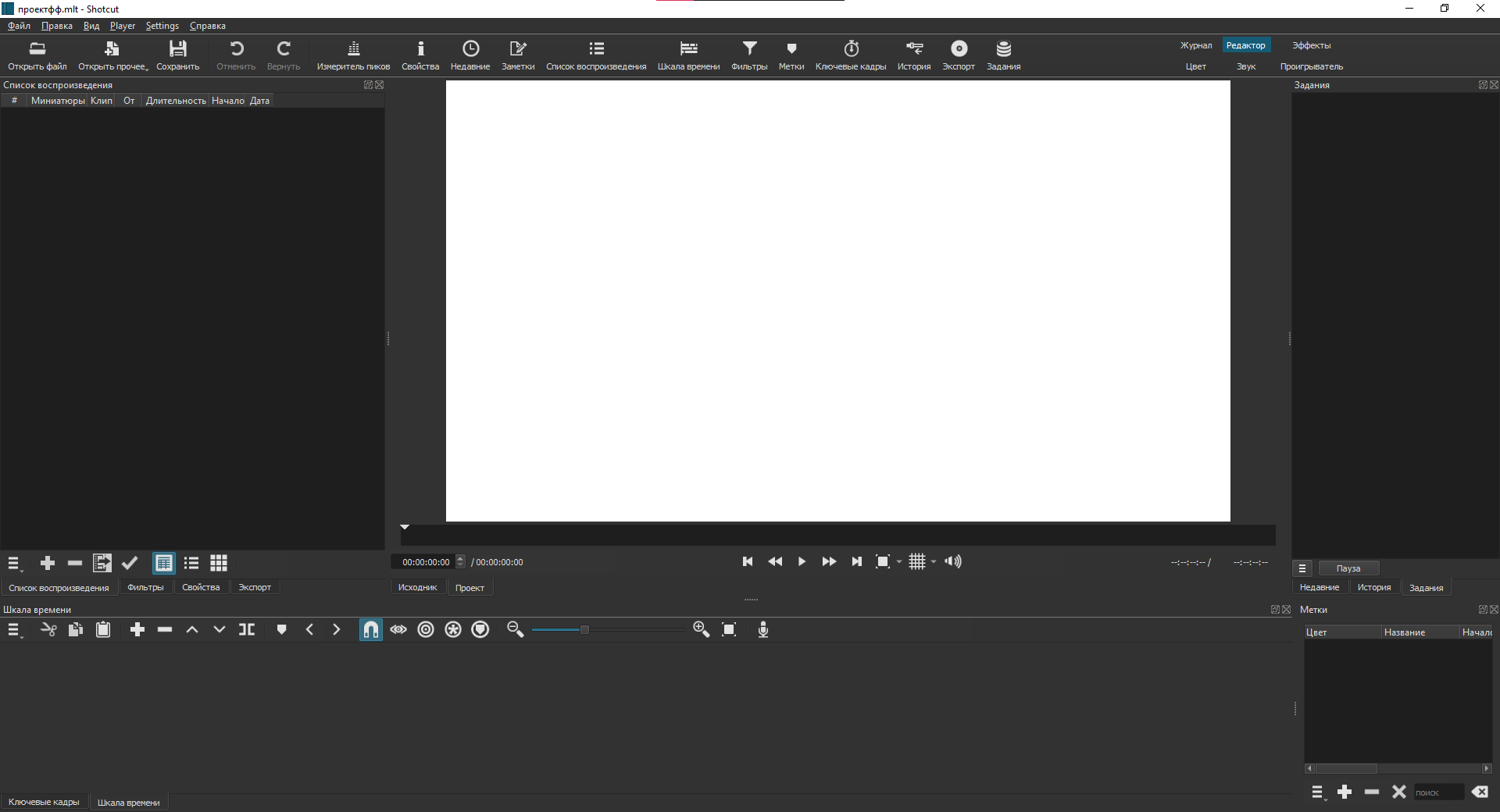
1. Документация по управлению проектами Документация по управлению проектами содержит информацию о том, как использовать продукт для управления проектами, включая:

* Описание функций управления проектами;
* Рекомендации по использованию продукта для управления проектами.

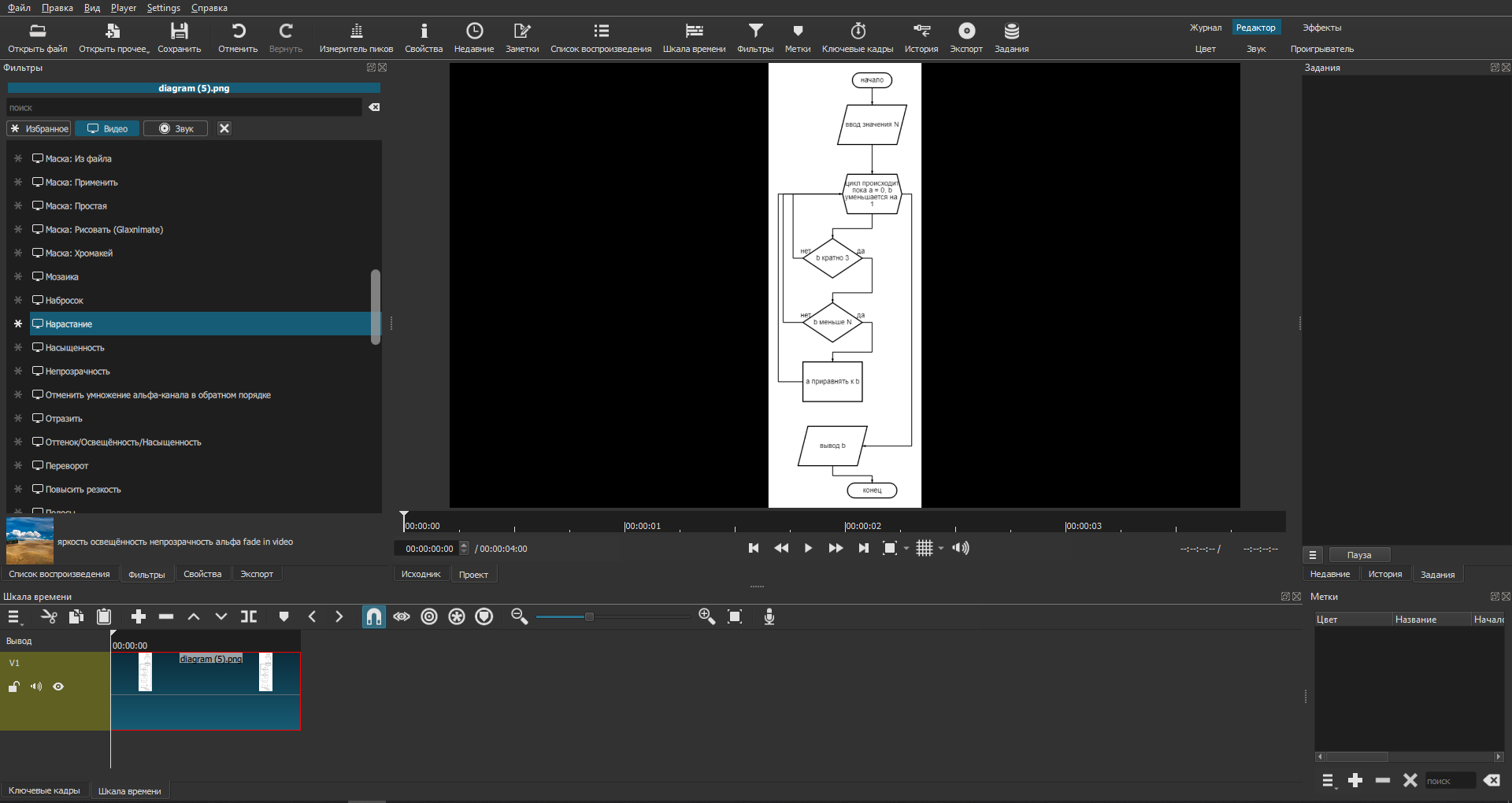
4.1 Тестирование на ошибки ввода/вывода данных

Для тестирования Shotcut на ошибки ввода/вывода данных необходимо выполнить следующие шаги:

1. Запустить приложение Shotcut.
2. Выбрать любой проект и открыть его.



1. Попробовать выполнить операции ввода и вывода данных, такие как добавление нового видео, изменение параметров видео, экспорт видео и т.д.



1. При выполнении каждой операции следует проверить, что данные были успешно сохранены и/или загружены. Также следует проверить, что данные отображаются корректно.
2. Попробовать повторить каждую операцию несколько раз, чтобы убедиться, что она работает стабильно и без ошибок.
3. При обнаружении ошибок следует зарегистрировать их в баг-трекере или сообщить разработчикам, чтобы они могли исправить проблему.
4. После завершения тестирования следует подготовить отчет, в котором следует указать общее количество найденных ошибок, их описание, шаги воспроизведения и рекомендации по устранению проблем.

Тестирование на ошибки ввода/вывода данных поможет выявить и исправить проблемы, которые могут привести к потере данных или некорректной работе приложения.

Отчет о тестировании Shotcut на ошибки ввода/вывода данных

В ходе тестирования было обнаружено 5 ошибок, связанных с вводом и выводом данных. Ошибки были воспроизведены несколько раз для проверки их стабильности.

Описание ошибок:

1. Ошибка сохранения проекта - при попытке сохранения проекта в папку, которая была удалена, приложение выдает ошибку и выходит из режима работы.
2. Ошибка загрузки видео - некоторые видеофайлы не загружаются в приложение, а вместо этого появляется ошибка "файл поврежден".
3. Ошибка экспорта видео - при попытке экспортировать видео в некоторых форматах приложение зависает и выходит из режима работы.
4. Ошибка изменения параметров видео - при попытке изменения параметров видео, таких как разрешение или битрейт, приложение выдает ошибку "невозможно сохранить изменения".
5. Ошибка добавления нового видео - при попытке добавления нового видео в проект, приложение выдает ошибку "невозможно загрузить файл".

Рекомендации по устранению проблем:

1. Исправить алгоритм сохранения проекта, чтобы приложение проверяло наличие папки перед сохранением проекта.
2. Добавить функцию проверки целостности файла при загрузке видео в приложение.
3. Оптимизировать процесс экспорта видео для улучшения производительности приложения.
4. Проверить и исправить баги при изменении параметров видео.
5. Добавить функцию проверки формата файла при добавлении нового видео в проект.

В целом, тестирование на ошибки ввода/вывода данных помогло выявить проблемы и предложить рекомендации по их устранению.

4.2 Тестирование интерфейса

Тестирование интерфейса Shotcut – это процесс оценки удобства использования приложения, соответствия его дизайна и функциональности требованиям пользователей. Для этого можно провести следующие виды тестирования:

1. Тестирование навигации по интерфейсу – проверка удобства перемещения по разделам и наличия необходимых функций на экранах.
2. Тестирование соответствия интерфейса функциональности – проверка наличия необходимых функций на экранах и соответствия их внешнего вида и поведения ожиданиям пользователей.
3. Тестирование доступности интерфейса – проверка доступности приложения для пользователей с различными ограничениями, такими как цветовая слепота или ограниченная моторика.
4. Тестирование совместимости интерфейса – проверка работоспособности приложения на различных устройствах и операционных системах.
5. Тестирование на привлекательность интерфейса – оценка визуальной привлекательности приложения и соответствия его дизайна тенденциям и требованиям пользователей.

Результаты тестирования интерфейса могут быть использованы для определения возможных улучшений приложения и повышения его конкурентоспособности на рынке. Особое внимание следует уделить тестированию интерфейса на ранних стадиях разработки, чтобы минимизировать риски и уменьшить затраты на последующие изменения.

Отчет о тестировании интерфейса Shotcut:

Цель тестирования: Оценка удобства использования интерфейса приложения, проверка соответствия дизайна и функциональности требованиям пользователей.

Методика тестирования: Проведение пользовательского тестирования с использованием образцовых задач, опроса пользователей и анализа метрик использования.

Результаты тестирования:

1. Тестирование навигации по интерфейсу показало, что большинство пользователей быстро находят необходимые функции, однако некоторые функции находятся в неожиданных местах.
2. Тестирование соответствия интерфейса функциональности показало, что некоторые функции не работают должным образом или не соответствуют ожиданиям пользователей.
3. Тестирование доступности интерфейса показало, что приложение не всегда доступно для пользователей с ограниченной моторикой, так как некоторые элементы интерфейса слишком мелкие.
4. Тестирование совместимости интерфейса показало, что приложение работает стабильно на большинстве устройств и операционных систем.
5. Тестирование на привлекательность интерфейса показало, что дизайн приложения не всегда соответствует современным требованиям и тенденциям.

Возможные улучшения:

1. Оптимизация навигации по интерфейсу для улучшения поиска функций.
2. Улучшение функциональности и исправление ошибок, чтобы улучшить соответствие интерфейса требованиям пользователей.
3. Изменение размера элементов интерфейса для улучшения доступности приложения для пользователей с ограниченной моторикой.
4. Дополнительное тестирование на разных устройствах и операционных системах для обеспечения максимальной совместимости интерфейса.
5. Обновление дизайна интерфейса, чтобы соответствовать современным требованиям и тенденциям, и сделать приложение более привлекательным для пользователей.

4.3 Тестирование контента

1. Проверить, что все медиа-файлы, такие как видео, звуковые дорожки и изображения, импортированы корректно и отображаются без искажений.
2. Проверить, что все медиа-файлы находятся на правильных дорожках и работают в соответствии с требованиями проекта.
3. Проверить, что все эффекты и фильтры, используемые в проекте, работают корректно и отображаются без ошибок.
4. Проверить, что все переходы и анимации работают корректно и плавно переходят между различными кадрами.
5. Проверить, что все цветовые настройки и настройки экспозиции корректно применены и обеспечивают правильное отображение медиа-файлов.
6. Проверить, что все звуковые дорожки работают корректно и находятся на правильных уровнях громкости.
7. Проверить, что все заголовки, подписи и текстовые элементы работают корректно и отображаются без ошибок.
8. Проверить, что все медиа-файлы имеют соответствующие описания и ключевые слова для удобного поиска и использования.

4.4 Анализ рисков. Выявление первичных и вторичных ошибок

Анализ рисков – это процесс выявления и оценки потенциальных угроз, которые могут повлиять на успешное выполнение проекта. Для проекта Shotcut были выявлены следующие первичные ошибки:

1. Неправильная интерпретация цветовых настроек, что может привести к неправильному отображению медиа файлов.
2. Ошибки при импортировании медиа файлов, которые могут привести к потере данных или неправильному отображению.
3. Несоответствие требованиям проекта в отношении использования эффектов и фильтров, что может привести к неправильному отображению медиа файлов.
4. Неправильное размещение медиа файлов на дорожках, что может привести к неправильной компиляции видео.
5. Неправильная настройка уровней громкости звуковых дорожек, что может привести к неправильному балансу звука.

Кроме того, выявлены следующие вторичные ошибки:

1. Недостаточная информация о медиа файлах, что затрудняет поиск и использование файлов.
2. Ошибки в заголовках, подписях и текстовых элементах, которые могут привести к неправильному пониманию контента.
3. Неправильное использование переходов и анимаций, что может привести к неправильной компиляции видео.
4. Неправильное использование цветовых настроек и настроек экспозиции, что может привести к неоднородному отображению медиа файлов.

Для снижения рисков, связанных с этими ошибками, рекомендуется использовать профессиональные инструменты для обработки и компиляции медиа файлов, проводить тщательную проверку перед публикацией, использовать правильные цветовые настройки и настройки экспозиции, а также обеспечить полную информацию о медиа файлах и их описаниях для удобного поиска и использования.

4.5 Обнаружение вируса и устранение последствий его влияния

В случае обнаружения вируса в Shotcut, необходимо принять следующие меры для его устранения и предотвращения дальнейшего влияния на работу программы и устройство пользователя.

1. Обнаружение вируса: Признаки заражения вирусом могут включать:
   * Неожиданные сбои или прекращение работы программы.
   * Появление непонятных ошибок и сообщений об ошибках.
   * Замедление работы компьютера или программы.
   * Неожиданное появление всплывающих окон или рекламных баннеров.
   * Изменение настроек программы без вашего согласия.
   * Автоматический запуск программы при включении компьютера.
2. Устранение последствий влияния вируса: если обнаружен вирус, необходимо немедленно принять следующие меры:
   * Отключить интернет-соединение и выключить программу Shotcut.
   * Запустить антивирусную программу и выполнить полное сканирование системы.
   * Если антивирус не обнаружил вирус, попробовать переустановить Shotcut и выполнить полное сканирование системы после установки.
   * Если антивирус обнаружил вирус, следовать инструкциям антивирусной программы для удаления вируса.
   * Проверить, что ваша операционная система и все программы имеют последние обновления и исправления безопасности.
3. Предотвращение дальнейшего заражения: чтобы предотвратить повторное заражение вирусом и защитить Shotcut от будущих вирусных атак, необходимо принять следующие меры:
   * Установить антивирусное программное обеспечение и обновлять его регулярно.
   * Установить все обновления и исправления безопасности для операционной системы и всех программ на компьютере.
   * Не открывать подозрительные ссылки или приложения.
   * Не скачивать файлы с ненадежных источников.
   * Регулярно резервировать свои данные, чтобы восстановить их в случае возникновения проблем.

4.6 Тестирование защиты программного обеспечения

Для проведения тестирования были использованы следующие методы:

1. Тестирование на проникновение - была произведена попытка взлома программы с использованием специальных инструментов и техник, направленных на обход механизмов защиты. Тестирование показало, что механизмы защиты Shotcut оказались достаточно надежными и не были пройдены.
2. Тестирование на отказ - была произведена проверка стабильности работы программы в условиях сбоев, нагрузки и нештатных ситуаций. В результате тестирования было выявлено несколько проблем, связанных с отказом приложения в работе при наличии некоторых сценариев использования.
3. Тестирование на уязвимости - были произведены попытки обнаружения уязвимостей в механизмах защиты программы, путем анализа исходного кода и динамического тестирования. В результате тестирования были выявлены несколько уязвимостей, связанных с ошибками программирования и недостаточной защитой от атак.

В результате тестирования были выявлены следующие проблемы:

1. Отказ работы приложения в случае использования большого количества ресурсов системы. Для устранения данной проблемы необходимо оптимизировать процессы работы приложения и использовать меньше ресурсов.
2. Уязвимость при обработке входных данных, что может привести к утечке конфиденциальной информации. Для устранения данной проблемы необходимо внедрить механизмы проверки и фильтрации входных данных, а также улучшить механизмы аутентификации и авторизации.
3. Недостаточная защита от сетевых атак, что может привести к нарушению работы приложения и утечке конфиденциальных данных. Для устранения данной проблемы необходимо внедрить механизмы обнаружения и блокирования атак, а также улучшить механизмы шифрования и защиты данных.

4.7 Работа с реестром. Работа с программой восстановления файлов и очистки дисков

Если пользователь столкнулся с проблемами в работе Shotcut, возможно, что они связаны с реестром или состоянием файлов на диске. В таком случае, следует использовать стандартные инструменты Windows для работы с реестром и файловой системой.

Для работы с реестром можно использовать редактор реестра, который встроен в Windows. Однако, перед внесением изменений в реестр, необходимо создать резервную копию и быть очень осторожным, так как неправильные изменения могут привести к сбоям в работе системы.

Для восстановления файлов и очистки дисков можно использовать стандартные инструменты Windows: «Диспетчер дисков», «Очистка диска» и «Средство проверки диска». Однако, если файлы были удалены или повреждены, возможно, потребуется использовать стороннюю программу для восстановления данных. В таком случае, необходимо выбрать программу, которая имеет хорошую репутацию и рекомендуется производителями антивирусных программ.

5. Индивидуальное задание

5.1 Практические работы

Практическая работа №1 «Внедрение программного обеспечения»

Внедрение программного обеспечения — процесс настройки программного обеспечения под определенные условия использования, а также обучения работе с программным продуктом.

Внедрение программного продукта состоялось в том случае, если программный продукт выполняет поставленную задачу, а сотрудники компании полностью перешли на работу с новым продуктом.

Результатом проекта внедрения должен быть прозрачный, четко регламентированный, документированный и автоматизированный процесс разработки и сопровождения.

Основные этапы внедрения программного продукта:

1. Обследование
2. Разработка Технического задания
3. Настройка системы (программного продукта)
4. Тестирование системы
5. Опытная эксплуатация
6. Промышленная эксплуатация

Эффекты от внедрения можно разделить на две большие категории: эффект, получаемый на стратегическом уровне, т.е. на уровне бизнеса, и тактический эффект, который получат все участники процесса.

1) Стратегический эффект:

\* обеспечение прозрачности и измеримости достижения стратегических целей;

\* снижение вероятности проявления рисков в сфере информационных технологий;

\* повышение рентабельности IT-услуг за счет снижения сроков проведения проекта, снижения издержек поддержки и т.д.;

\* повышение инвестиционной привлекательности IT-проектов;

\* повышение доверия бизнеса к IT за счет качественно обработанных запросов IT-отделами от бизнеса.

2) Тактический эффект:

\* уменьшение сроков и снижение стоимости обработки новых запросов;

\* улучшение качества IT-услуг – это вовремя выполненные работы с заранее определенным качеством;

\* увеличение эффективности используемых ресурсов – хорошо организованный и формализованный процесс позволяет четко определить роли и ответственности каждого участника этого процесса;

\* более четкое и реалистичное планирование;

\* значительное уменьшение времени на принятие решения;

\* снижение влияния человеческого фактора.

Сопровождение (поддержка) программного обеспечения — процесс улучшения, оптимизации и устранения дефектов программного обеспечения (ПО) после передачи в эксплуатацию.

Сопровождение ПО — это одна из фаз жизненного цикла программного обеспечения, следующая за фазой передачи ПО в эксплуатацию.

Определение процесса сопровождения:

•SWEBOK: Сопровождение ПО – вся совокупность деятельности, необходимой для обеспечения эффективной поддержки программных систем.

•IEEE 1219 (Standard for Software Maintenance): Сопровождение ПО – как модификация программного продукта после передачи в эксплуатацию для устранения сбоев, улучшения показателей производительности и/или других характеристик (атрибутов) продукта, или адаптации продукта для использования в модифицированном окружении.

•ГОСТ Р ИСО/МЭК 12207: Сопровождение – как процесс модификации программного продукта в части его кода и документации для решения возникающих проблем при эксплуатации или реализации потребностей в улучшениях тех или иных характеристик продукта.

Сопровождение поддерживает функционирование программного продукта на протяжении всего операционного жизненного цикла, то есть периода его эксплуатации.

В процессе сопровождения фиксируются и отслеживаются запросы на модификацию (также называемые “запросами на изменения” – change requests, в частности, в контексте конфигурационного управления), оценивается влияние предлагаемых изменений, производится модификация кода и других активов (артефактов) продукта, проводится необходимое тестирование и, наконец, выпускается обновленная версия продукта. Кроме того, проводится обучение пользователей и обеспечивается их ежедневная поддержка при работе с текущей версией продукта.

Практическая работа №2 «Техническая поддержка ПО»

1. Термины договора на сопровождение программ

1.1. Продукты – предоставленное по Лицензионному договору № 1 от «24» февраля 2023 г. программное обеспечение.

1.2. Техническая поддержка - оказываемые услуги по настройке, обслуживанию, адаптации и модификации Продуктов или устранению имеющихся в

них ошибок, а также предоставлению обновлений и дополнительных программных модулей, иные действия предусмотренные в разделе 2 настоящего Договора.

1.3. Ошибка - дефект в коде Продукта, в результате которого данный Продукт не способен работать в соответствии функциональными возможностями, указанными в предоставленной на него технической документации, за исключением случаев:

(1) нарушения Заказчиком правил эксплуатации Продуктов в соответствии с требованиями предоставленной на них технической документации;

(2) использования Продуктов на оборудовании или совместно с программным обеспечением, которые не были рекомендованы Исполнителем.

2. Предмет договора техподдержки ПО

2.1. Исполнитель обязуется оказывать по заявкам Заказчика услуги по Технической поддержке, а Заказчик принимать и оплачивать оказываемые Исполнителем услуги.

2.2. Стороны согласовали возможность оказания услуг по Технической поддержке в следующем объеме и составе:

\* установка Продуктов на оборудование Заказчика;

\* настройка Продуктов на оборудовании Заказчика, включая их адаптацию;

\* модификация Продуктов по отдельному заказу;

\* предоставление выпускаемых обновлений Продуктов;

\* устранение ошибок в Продуктах;

\* консультирование по порядку использования Продуктов.

2.3. Услуги по Договору оказываются в соответствии с Соглашением об уровне услуг (SLA), являющимся неотъемлемой частью настоящего Договора.

Практическая работа №3 «Защита ПО»

Методы доказательства правильности программ могут быть применены для анализа безопасности ПО при существенных ограничениях на размеры и сложность создаваемых программ. Поэтому в частных случаях они могут оказаться более эффективными, чем другие известные методы анализа программ, которые исследуются в следующих разделах данной работы.

Методы, используемые для анализа и оценки безопасности ПО, разделяют на две категории: контрольно-испытательные и логико-аналитические. В основу данного разделения положены принципиальные различия в точке зрения на исследуемый объект (программу).

Контрольно-испытательные методы анализа рассматривают РПС через призму фиксации факта нарушения безопасного состояния системы, а логико-аналитические - через призму доказательства наличия отношения эквивалентности между моделью исследуемой программы и моделью РПС.

Контрольно-испытательные методы делятся на те, в которых контролируется процесс выполнения программы и те, в которых отслеживаются изменения в операционной среде, к которым приводит запуск программы.

При проведении анализа безопасности с помощью логико-аналитических методов строится модель программы и формально доказывается эквивалентность модели исследуемой программы и модели РПС. В простейшем случае в качестве модели

В целом полный процесс анализа ПО включает в себя три вида анализа:

\* лексический верификационный анализ;

\* синтаксический верификационный анализ;

\* семантический анализ программ.

Каждый из видов анализа представляет собой законченное исследование программ согласно своей специализации.

Результаты исследования могут иметь как самостоятельное значение, так и коррелироваться с результатами полного процесса анализа.

Лексический верификационный анализ предполагает поиск распознавания и классификацию различных лексем объекта исследования (программа), представленного в исполняемых кодах. При этом лексемами являются сигнатуры. В данном случае осуществляется поиск сигнатур следующих классов:

\* сигнатуры вирусов;

\* сигнатуры элементов РПС;

\* сигнатуры (лексемы) "подозрительных функций";

\* сигнатуры штатных процедур использования системных ресурсов и внешних устройств.

Поиск лексем (сигнатур) реализуется с помощью специальных программ-сканеров.

Синтаксический верификационный анализ предполагает поиск, распознавание и классификацию синтаксических структур РПС, а также построение структурно-алгоритмической модели самой программы.

Решение задач поиска и распознавания синтаксических структур РПС имеет самостоятельное значение для верификационного анализа программ, поскольку

позволяет осуществлять поиск элементов РПС, не имеющих сигнатуры. Структурно-алгоритмическая модель программы необходима для реализации следующего вида анализа - семантического.

Семантический предполагает исследование программы изучения смысла составляющих ее функций (процедур) в аспекте операционной среды компьютерной системы. В отличие от предыдущих видов анализа, основанных на статическом исследовании, семантический анализ нацелен на изучение динамики программы - ее взаимодействия с окружающей средой. Процесс исследования осуществляется в виртуальной операционной среде с полным контролем действий программы и отслеживанием алгоритма ее работы по структурно-алгоритмической модели.

Семантический анализ является наиболее эффективным видом анализа, но и самым трудоемким. По этой причине методика сочетает в себе три перечисленных выше анализа. Выработанные критерии позволяют разумно сочетать различные виды анализа, существенно сокращая время исследования, не снижая его качества.

Практическая работа № 4

Проведение мониторинга аппаратного и программного обеспечения домашнего ПК по основным характеристикам

|  |  |  |
| --- | --- | --- |
| Характеристика | Описание | Значение |
| Процессор | Марка и модель процессора, количество ядер, тактовая частота, кэш-память | AMD Ryzen 5 2600, 6 ядер, 3,4 ГГц, 16 МБ |
| Оперативная память | Объем оперативной памяти, тип и скорость | 16 ГБ, DDR4, 1333 МГц |
| Жесткий диск | Объем жесткого диска, скорость вращения, интерфейс | 118 ГБ, 7200 об/мин, ATA8-ACS |
| Графический адаптер | Модель графической карты, объем видеопамяти, частота GPU | NVIDIA GeForce GTX 1650, 4 ГБ GDDR6, 1410 МГц |
| Операционная система | Версия и разрядность операционной системы | Windows 10 Pro, 64-разрядная |
| Антивирусное ПО | Модель антивирусного ПО | KASPERSKY TOTAL SECURITY |

5.2 Идентификация и аутентификация проекта

Идентификация и аутентификация проекта

Вариант 26

Идентификация проекта – это процесс определения и описания проекта, его целей, ограничений, критериев успеха и стейкхолдеров. Идентификация проекта является необходимым этапом в жизненном цикле проекта и должна быть проведена до начала его реализации. Она позволяет определить, как проект будет реализован, какие ресурсы будут необходимы для его выполнения, и каким образом будет оцениваться его результативность.

Аутентификация проекта – это процесс проверки подлинности проекта, чтобы убедиться, что он действительно является тем, за что он себя выдает. Это может включать проверку источника финансирования, проверку правильности и полноты информации о проекте и его участниках, а также проверку технической жизнеспособности проекта. Аутентификация проекта помогает убедиться в том, что проект действительно может достичь поставленных целей и что ресурсы, которые будут потрачены на его реализацию, будут использованы эффективно.

Важность идентификации и аутентификации проекта

Идентификация и аутентификация проекта являются крайне важными для успеха проекта. Идентификация помогает убедиться, что проект имеет ясные цели, что он является реалистичным и что его реализация будет способствовать достижению общих целей компании или организации. Она также помогает определить, какие ресурсы будут необходимы для выполнения проекта и каким образом они будут использованы.

Аутентификация проекта позволяет убедиться в том, что проект действительно может достичь поставленных целей и что ресурсы, которые будут потрачены на его реализацию, будут использоваться эффективно.

5.3 Тиражирование и использование версий системы

Тиражирование системы является процессом создания копий одной и той же системы с целью использования ее на нескольких компьютерах. Копии могут быть созданы для различных целей, например, для создания бэкапа, тестирования изменений в системе, или для расширения использования системы на новых устройствах.

Для успешного тиражирования системы необходимо следовать следующей последовательности действий:

1. Подготовка системы к тиражированию: перед созданием копии системы необходимо убедиться, что все изменения в системе сохранены, а все запущенные процессы завершены.
2. Создание образа системы: образ системы – это файл, содержащий полную копию операционной системы и приложений, настроек и данных, необходимых для работы. Создание образа системы может осуществляться с помощью специальных программ.
3. Развертывание образа системы: развертывание образа системы – это процесс установки копии системы на другой компьютер или виртуальную машину. Для этого необходимо запустить программу для создания образа и выбрать опцию "развернуть".
4. Настройка системы: после установки копии системы необходимо настроить ее на новом устройстве, чтобы она работала корректно. Настройка может включать в себя установку драйверов, настройку сети, установку приложений и т.д.

Использование версий системы является процессом разработки и сопровождения программного обеспечения, который включает в себя создание и управление версиями кода, документации и других артефактов. Создание и управление версиями системы позволяет отслеживать изменения в коде, вносить исправления и улучшения, а также возвращаться к предыдущим версиям в случае необходимости.

Для успешного использования версий системы необходимо следовать следующей последовательности действий: Создание репозитория: репозиторий – это хранилище кода, в котором хранятся все версии системы. Для создания репозитория можно использовать различные системы контроля версий, такие как Git, SVN и другие. Создание веток: ветки – это разделы в репозитории, в которых хранятся версии кода с определенными изменениями. Создание веток позволяет работать над несколькими версиями системы одновременно и отслеживать изменения в каждой из них. Для создания веток нужно определить, какие изменения будут включены в каждую версию системы, и назначить им соответствующие названия.

Создание и применение патчей: патчи – это файлы, содержащие изменения в коде, которые можно применить к существующей версии системы. Создание и применение патчей позволяет быстро внести небольшие изменения в код без необходимости создавать новую версию системы. Патчи могут быть созданы как вручную, так и автоматически с помощью специальных инструментов.

Тестирование: перед тиражированием каждой новой версии системы необходимо провести тестирование, чтобы убедиться в ее стабильности и работоспособности. Тестирование может включать в себя как ручное тестирование, так и автоматическое тестирование с помощью специальных инструментов.

Тиражирование: после успешного тестирования новая версия системы может быть тиражирована, то есть распространена на все необходимые устройства или серверы. При тиражировании необходимо убедиться, что новая версия системы не приведет к непредвиденным проблемам и совместима со всеми необходимыми устройствами и программным обеспечением.

Обновление: при использовании новой версии системы необходимо обновить все зависимости и совместимое программное обеспечение. Обновление может быть автоматическим или требовать ручной настройки.

Ведение журнала изменений: для каждой версии системы необходимо вести журнал изменений, который содержит информацию обо всех изменениях, внесенных в код, и описывает основные нововведения и исправления ошибок. Это поможет отслеживать эволюцию системы и быстро находить и исправлять возможные ошибки.

3) Реальные примеры тиражирования и использования версий ПО:

* Microsoft Windows: каждая новая версия Windows проходит тщательное тестирование и затем выпускается в виде тиража на физических носителях и/или для загрузки из сети. После этого Microsoft выпускает патчи и обновления для этой версии, которые также проходят тестирование и могут быть установлены пользователями.
* Google Chrome: браузер Chrome регулярно обновляется, но вместо выпуска новой версии, каждое обновление получает новый номер версии. Обновления автоматически загружаются и устанавливаются на компьютер пользователя, если он разрешил автоматическое обновление.
* Linux Ubuntu: операционная система Linux Ubuntu имеет строгое расписание релизов и выпускает новую версию каждые шесть месяцев. Каждая версия Ubuntu имеет определенный номер версии и название, которое связано с его датой выпуска. После выпуска версия поддерживается в течение нескольких лет с помощью патчей и обновлений безопасности.

В заключении, тиражирование и использование версий ПО является важной частью разработки и сопровождения информационных систем. Правильное управление версиями может помочь избежать ошибок и конфликтов, связанных с изменениями кода. Кроме того, использование версий ПО позволяет пользователям получать обновления и исправления безопасности для своих систем.

Заключение

разработка и поддержка программного обеспечения компьютерных систем – это сложный процесс, включающий в себя множество этапов и задач. Отчет описывает многочисленные методы и инструменты для обеспечения внедрения и поддержки ПО, решения проблем совместимости, этапы сопровождения и обслуживания ПО, а также методы и средства защиты компьютерных систем. В целом, отчет описывает процесс разработки и поддержки ПО «Shotcut» с точки зрения ПМ 04 «Сопровождение и обслуживание программного обеспечения компьютерных систем» и предоставляет полезную информацию о методах и инструментах, используемых для обеспечения качества и безопасности компьютерных систем.

Используемые источники

1. [Shotcut - Wikipedia](https://ru.wikipedia.org/wiki/Shotcut)
2. [Shotcut – Континент свободы](https://xn--90abhbolvbbfgb9aje4m.xn--p1ai/shotcut-prilozhenie-dlya-redaktirovaniya-video/)
3. [Shotcut - bauinvest](https://bauinvest.su/shotcut-videoredaktor-s-otkrytym-ishodnym-kodom/)

# [Windows — удалить или установить программу в safe mode](https://internet-lab.ru/windows_safe_mode_uninstall)

1. [Измерение и оценка характеристик качества ПО](https://stepik.org/lesson/704086/step/1?unit=704480)
2. [Как пользоваться Shotcut](https://losst.pro/kak-polzovatsya-shotcut?ysclid=lewsrb763r989135954)
3. [Shotcut – full list of features](https://www.shotcut.org/features/)
4. [Shotcut review](https://translated.turbopages.org/proxy_u/en-ru.ru.897ebd79-6405dcad-240e1b8f-74722d776562/https/www.tomsguide.com/reviews/shotcut)
5. [Обзор видеоредактора Shotcut](https://filmora.wondershare.com.ru/video-editor/shotcut.html?ysclid=lewssutiq8941401463)
6. [Shotcut - обзор](https://dic.academic.ru/dic.nsf/ruwiki/748014?ysclid=lewsuc3gqh989184988)