МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ, МОЛОДЕЖИ РЕСПУБЛИКИ КРЫМ

ОТДЕЛ ОБРАЗОВАНИЯ, МОЛОДЕЖИ И СПОРТА АДМИНИСТРАЦИИ ЧЕРНОМОРСКОГО РАЙОНА

**МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ «ЧЕРНОМОРСКАЯ СРЕДНЯЯ ШКОЛА №1 ИМ. НИКОЛАЯ КУДРИ» МУНИЦИПАЛЬНОГО ОБРАЗОВАНИЯ ЧЕРНОМОРСКИЙ РАЙОН РЕСПУБЛИКИ КРЫМ**

Создание веб-сайта

Работа обучающегося 9-А класса

Мельника Павла Юрьевича

Руководитель проекта

Ященко Николай Григорьевич

Работа допущена к защите «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 год

Подпись руководителя проекта \_\_\_\_\_\_ (\_\_\_\_\_\_)

Черноморское, 2021г.

Table of Contents

[1 Теоретическая часть 4](#__RefHeading___Toc747_1140880220)

Введение

Темой индивидуального проекта является «Планировщик заданий». Реализация данной темы важна в наше время, так как сейчас у людей очень много дел, которые нужно запомнить или записать, а также организовать эту информацию. Бумажные носители можно смело назвать устаревшими, так как у них имеется множество недостатков:

* Расход конечных природных ресурсов на бумагу и принадлежности для письма
* Необходимость красиво писать от руки
* Сложность быстрого обмена информацией
* Громоздкость при наличии больших объемов данных
* Сложность поиска среди доступной информации

В то же время у хранения данных в электронном виде множество преимуществ:

* Информацию можно легко и быстро передавать в любую точку мира через интернет
* Можно осуществлять поиск нужной информации с помощью компьютерных программ, удобно структурировать данные и выполнять их анализ
* Хранение большого количества информации не занимает много места в реальном мире, не требует расхода природных ресурсов.
* ременную карту памяти размером с ноготь можно поместить содержимое всех библиотек мира
* Данные в электронном виде легко изменять и обновлять, в то время как данные на бумаге после первой же ошибки нужно переписывать заново, если нужно созранить приемлемый вид документа

Поэтому данное приложение будет довольно востребованным. По сравнению с другими похожими проектами можно выделить следующие отличительные черты: очень простой, понятный и не надоедливый дизайн, выделение прошедших и будущих задач, сбор статистики о выполнении задач в срок

Цель работы – изучение технологий разработки программ для персональных компьютеров, а также разработка приложения "Список задач"

Задачи:

1. Определить актуальность темы
2. Изучить и сравнить способы и технологий разработки компьютерных приложений
3. Научиться разрабатывать приложения для ПК
4. Разработать приложение "Список задач"

# Теоретическая часть

План работы над проектом:

1. Спецификация. На этом этапе происходит определение требований к результату проекта т.е. к разрабатываемому приложению
2. Определение технологии выполнения работы
3. Обучение. На этом этапе будет проводиться изучение обучающих материалов в области разработки компьютерных приложений.
4. Программирование. После проведения спецификации и обучения программированию, будет создан готовый продукт при помощи языков программирования.
5. Отладка. На данном этапе программист занимается отладкой программы, то есть поиском и устранением ошибок.
6. Тестирование. Тестирование программы очень важно, поскольку в большинстве случаев программисты создают их не для личного применения, а для того, чтобы их программой пользовались другие. На этапе тестирования разработчик проверяет поведение разрабатываемого продукта на различных устройствах и операционных системах при различных наборах данных.
7. Сопровождение, внесение изменений, оптимизация. После запуска программы в эксплуатацию осуществляется её сопровождение, т.е. внесение изменений на основе выявленных недочетов в процессе эксплуатации продукта, а также проводится оптимизация функционала или добавление нового

Для разработки программы был выбран язык С\# - читается как «Си шарп» - это язык программирования от компании Microsoft. Вообще его решили разработать для проектов под Windows, но не так давно данный язык стал универсальным. Сейчас на этом языке пишут различные игры, десктопные программы, веб-сервисы, нейросети и графику для метавселенных. Разрабатывал язык знаменитый Андерс Хейлсберг.

Преимущества языка С#:

* Независим от железа. Разработчику не потребуется адаптировать приложение под различные платформы и системы. За человека это будет выполнять специальная виртуальная машина, которая уже вшита NET Framework. Получается, что один и тот же код удастся запускать на различных гаджетах. Например, на телефонах, серверах, компьютерах и т.д.
* Хорошая совместимость с Windows. Не зря разработчики делали данный язык именно в Microsoft.
* Разрешается управлять памятью. Для того чтобы приложение хорошо работало и без лагов, её требуется время от времени очищать от различных ссылок, кэша, ненужных объектов и т.д. В С\# этот процесс осуществляется автоматическим путем. Специалисту не нужно будет следить за тем, как расходуется память, бороться с её утечками или убирать мертвые куски кода.
* Строжайшая типизация. Как только специалист объявит переменную в С\#, потребуется сначала указать, что именно в ней лежит – число, массив или строчка. Таким способом придется немного дольше разрабатывать, но ваш код будет работать правильно и предсказуемо. Числа будут взаимодействовать с числами, строки со строками и т.д. В языках, у которых слабая типизация свободы и драйва будет больше, но специалист может случайно пропустить ошибку. Она потом всплывет в готовом приложении и приведет к серьезным последствиям.
* Огромное сообщество. На данном языке пишут множество специалистов во всем мире. В социальных сетях и различных сообществах можно задавать разные вопросы, обсуждать сложные темы или искать готовые решения. Некоторым программистам удается находить ментора, который может поделиться собственными знаниями и поможет намного быстрее выучить язык С\#.
* Синтаксический сахар. В программе С\# существует множество различных методов, как сократить код, чтобы не нарушить логику приложения. Опытные специалисты называют подобные приемы «синтаксическим сахаром» - они позволяют выполнить код намного понятнее, проще и красивее.

AvaloniaUI — кроссплатформенный фреймворк для разработки пользовательского интерфейса на основе языка разметки XAML. Достаточно молодой и активно развивающийся фреймворк для кроссплатформенной разработки, разработанный Стивеном Кирком и поддерживаемый .NET Foundation и сообществом. Он поддерживается на Windows через .NET, на Linux через Xorg и на macOS. Сама Avalonia основана на WPF/UWP. Код пишется на языке C\#. Благодаря XAML Avalonia позволяет создавать гибкие и стилизованные интерфейсы. Имеет хорошую документацию, хоть и не большое, но очень приветливое рускоязычное сообщество. Для Visual Studio существует расширение, которое содержит шаблоны проектов и элементов управления. С ними вы легко вникнете в суть и сразу сможете приступить к разработке интерфейсов.

AvaloniaUI является open-source проектом, который активно развивается и совершенствуется. Однако это не мешает уже использовать его в каких-либо крупных проектах.

Исходный код Avalonia полностью открыт — вы можете найти исходный код на официальном сайте Avalonia, выпущенной под лицензией MIT.

Avalonia написана на .NET Standard и на 100\% совместима со всеми версиями .NET Core и .NET Framework.

Авалония очень быстро развивается чрезвычайно умными людьми по всему миру, которые работают с клиентами и знают, что им нужно.

В Avalonia есть поддержка Visual Studio и Rider для создания файлов XAML и XAML intellisense, хотя Avalonia XAML intellisense все еще отстает от того, что может предложить WPF XAML intellisense.

Фреймворки WPF и Silverlight придумали ряд парадигм программирования, выходящих за рамки обычных парадигм программирования ООП, которые позволяют создавать код значительно чище и быстрее тем, кто понимает такие парадигмы.

Avalonia позволяет создавать приложения, которые выглядят и ведут себя одинаково на разных платформах, но также поддерживает настройку для конкретной платформы.

Несколько чрезвычайно популярных проектов с открытым исходным кодом .NET перешли с WPF на Avalonia, чтобы сделать свои продукты мультиплатформенными