

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра Математического и компьютерного моделирования

Проектирование и разработка программного приложения

«Текстовый редактор»

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студентки 4 курса 441 группы

направление 09.03.03 — Прикладная информатика

механико-математического факультета

Пеньковой Екатерины Андреевны

Научный руководитель
доцент, к.ф. — м.н.,

Ю. В. Мельникова

Зав. кафедрой
зав. каф., д.ф.-м.н., доцент

Ю. А. Блинков

Саратов 2023

Введение. В современном мире пользователи часто сталкиваются с проблемой хранения больших объемов текстовой информации. Например: номера телефонов, списки покупок, пароли. Вся эта информация хранится в лучшем случае на листке, который вероятнее всего в скором времени будет утерян. Чтобы не происходило подобного, на помощь приходят различные сервисы и хранилища.

Потребность в удобных приложениях разных направлений растет с каждым днем. Если возникает необходимость написать заметку, новое изученное слово или что-то другое, то пользователь обращается к различным текстовым редакторам, которые сохраняют необходимую информацию. Пользуясь таким приложением, пользователь может не волноваться о потере данных.

В 2020 году произошел массовый переход людей на удаленную работу в связи с пандемией. В 2022 году большое количество компаний разом покинули российский рынок ¹. Сложно посчитать точное число, так как практически ежедневно СМИ публикуют новости о том, что руководство той или иной компании принимает подобное решение. Этот переход оказался очень неожиданным, застав всех врасплох. Программное обеспечение очень сложно заменить отечественными аналогами, но есть выход. Например, вместо офисных программ Word и Excel можно использовать «Мой Офис». Частично или полностью ушли: Microsoft, Adobe, Nvidia, Avast, SAP, Oracle, Xerox, Figma. Санкционная повестка в отношении компаний, работающих в России, является основным стимулом для разработки собственных продуктов в сфере IT. Последняя волна международных запретов только укрепила эту тенденцию.

Переход российских компаний на отечественное программное обеспечение вскоре может стать не просто желательной, но откровенно вынужденной мерой. На данный момент существует большое количество блокнотов-приложений, с огромным количеством разнообразных функций, внешним видом и возможностями. Однако, исследования показали, что пользователь не желает разбираться в особенностях приложения и предпочитает молниеносно внести в память необходимую заметку. Основываясь на таком желании пользователя, приложение должно быть простым для использования, интуитивно понятным, креативным и привлекательным в графическом интерфейсе.

¹Указ Президента РФ от 30 марта 2022г. №166

Блокнот — это простейший текстовый редактор, который можно использовать в качестве удобного средства для создания, редактирования, просмотра текстовых файлов.

Объект работы — приложение «Блокнот».

Цель данной работы заключается в описании процессов проектирования, разработки и тестирования приложения для ведения заметок «Блокнот».

Данная цель определила следующие основные задачи работы:

- выявить предпочтения пользователей с помощью опроса;
- рассмотреть этапы разработки приложения;
- рассмотреть этапы тестирования приложения.

Среди тестировщиков существует огромное разделение на более узкие специальности. Например: тестирование безопасности, производительности, удобства использования. Но в самом широком смысле их можно разделить на ручных тестировщиков и тестировщиков-автоматизаторов. В данной работе рассматривается ручное тестирование.

Ручное тестирование – это тип тестирования программного обеспечения, при котором тестировщик вручную проводит тесты без помощи каких-либо средств автоматизации.

Структура работы. Основная часть состоит из 3 разделов:

- Обзор существующих программных решений и анализ предпочтений пользователей;
- Выбор необходимого программного обеспечения;
- Этапы разработки программного обеспечения.

В первом разделе собрана вся информация о текстовых редакторах и об особенностях их использования.

Также был проведен опрос респондентов с целью выяснения их мнения о разрабатываемом программном продукте. Для того чтобы опросить большее количество респондентов, была разработана анкета в Google Forms. Данная анкета была предложена студентам разного возраста. Результаты анкетирования использовались для выделения основных функций приложения. Пользователям были заданы два вопроса:

- Пользуетесь ли Вы приложениями для ведения заметок?

- Какие критерии важны для Вас при выборе приложения для ведения заметок?

Второй раздел относится к выбору необходимого программного обеспечения для разработки.

В качестве программного обеспечения используется программа PyCharm. Это кроссплатформенная интегрированная среда разработки для языка программирования Python, разработанная компанией JetBrains на основе IntelliJ IDEA., а также Tkinter – кроссплатформенная библиотека для разработки графического интерфейса на языке Python.

Существует много причин, объясняющих, почему библиотека tkinter стала в мире Python стандартом разработки переносимых графических интерфейсов. Такие преимущества, как доступность, переносимость, простота получения, документированность и наличие расширений, делают ее наиболее широко используемым решением для Python в области GUI в течение многих лет.

Создатель языка Python — нидерландский программист Гвидо ван Россум. Он был участником проекта по написанию ABC, языка для обучения программированию. В конце 1989 года Гвидо приступил к разработке нового языка и задумал его как потомка ABC, способного к обработке исключений и взаимодействию с операционной системой Amoeba. Так и получился Python.

Откуда такое название? Многие разработчики считают, что язык назван в честь семейства змей, но это не так. Когда Гвидо работал над проектом, он любил смотреть комедийное шоу «Летающий цирк Монти Пайтона», поэтому и нарёк своё творение в честь британской комик-группы. Так что правильно произносить название языка как «Пайтон».

Python свободно распространялся через интернет и со временем у него появились последователи — люди, заинтересованные в развитии этого языка программирования. Первая публикация Python состоялась в феврале 1991 года — это была версия 0.9.0.

Сегодня Python активно используется в веб-разработке наряду с такими языками, как PHP, JavaScript, Java. Почему был выбран язык Python?

Его основные свойства:

- Понятность кода. Синтаксическая особенность Python — выделение блоков кода отступами, что значительно упрощает зрительное восприятие программ, написанных на этом языке;
- Интерпретируемость. Программы, написанные на языке программирования Python, не переводятся в машинный код, а сразу выполняются программой-интерпретатором. Это позволяет запускать код на любой платформе с установленным заранее интерпретатором;
- Объектно ориентированность. Python — это язык, созданный согласно парадигме объектно ориентированного программирования (ООП). В ней основными являются понятия объекта и класса. Классы — это специальные типы данных, объекты — экземпляры классов. То есть любое значение является объектом конкретного класса. В Python вы можете не только использовать уже существующие классы, но и создавать свои собственные;
- Динамическая типизация. В отличие от С-подобных языков программирования, в Python переменные связываются с типом в момент присваивания в них конкретных значений.

Python является самым популярным языком программирования в рейтинге ТЮВЕ за август 2022 года. Его применяют повсеместно: в аналитике данных, тестировании и разработке игр. Кроме того, Python нашел широкое применение среди ученых благодаря своей простоте.

Несомненно, Python прекрасен со всех сторон. Но у всех есть свои недостатки. Ниже рассмотрим самые существенные из них:

- Python является одним из самых медленных языков программирования.
- Python не подходит для задач, которые требуют большого объема памяти.

В третьем разделе описаны этапы разработки приложения «Текстовый редактор».

Разработка приложения начиналась с проектирования. Сначала программа «Блокнот» проектировалась в голове, далее была перенесена на лист бумаги. Первый этап проектирования приложения состоял в том, что нужно было понять, из каких компонентов будет состоять программа. Окно долж-

но иметь три основные кнопки и иметь возможность изменяться в размере. Также основными компонентами окна должны быть: строка заголовка, меню верхнего уровня, панель инструментов для меню второго уровня и статусная панель.

Для создания меню верхнего уровня необходимо было тщательно продумать, какой пункт меню какие подпункты должен содержать и нарисовать схему меню.

Далее нужно было нарисовать программу со всеми составляющими её компонентами, включая меню второго уровня с пиктограммами ("Сохранить", "открыть" и "печать") и статусную панель.

В данной работе представлено объектно-ориентированного проектирование при помощи нотации UML. Язык UML представляет собой графический язык моделирования общего назначения, предназначенный для спецификации, визуализации, проектирования и документирования всех артефактов, создаваемых при разработке программных систем.

Также необходимо было спроектировать диаграмму прецедентов, отражающую отношения между актёрами и прецедентами. Прецеденты – это технология определения функциональных требований к системе. Работа прецедентов заключается в описании типичных взаимодействий между пользователями программного обеспечения и самой системой и предоставлении описания процесса ее функционирования.

После этапа проектирования идет этап разработки. Процесс разработки программного обеспечения — процесс, посредством которого потребности пользователей преобразуются в программный продукт.

Разработанная программа умеет создавать и открывать текстовые документы, форматировать и редактировать текст. Также должна быть предусмотрена возможность сохранения текста в файл.

Дизайн приложения интуитивно понятен и прост в использовании. В соответствии с рисунком 1 изображено разработанное приложение.

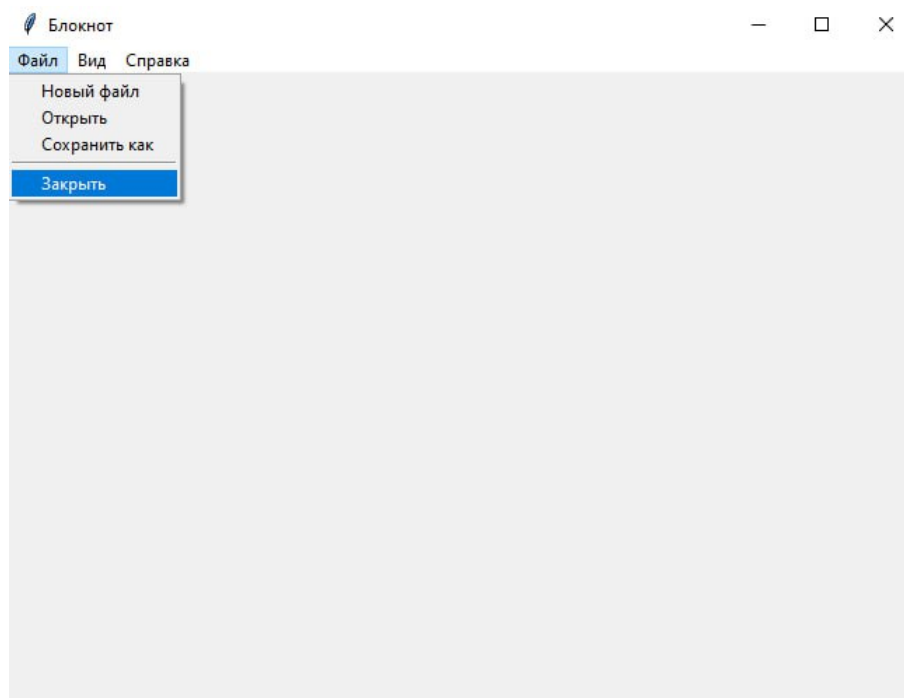


Рисунок 1 — Использование стандартных функций

Первоначально оно состоит из одного пустого окна с возможностью использовать стандартные функции: открыть, сохранить как, закрыть.

В соответствии с рисунком 2 продемонстрирована возможность выбора шрифта и цвета темы. По результатам многочисленных исследований, всегда выигрывает светлая тема. Единственное, что можно отметить: темная тема действительно снижает нагрузку на глаза в ночное время.

При нажатии на кнопку новый файл открывается новое окошко для новой записи, в соответствии с рисунком 3.

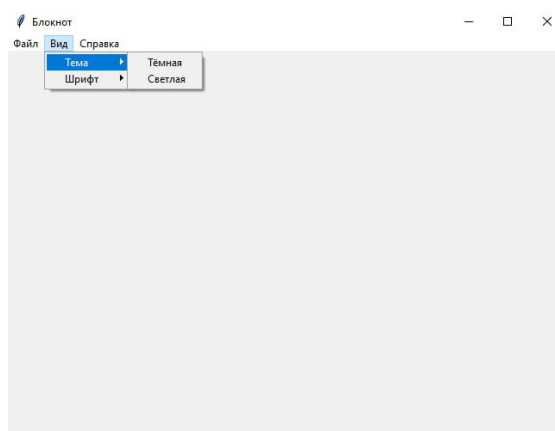


Рисунок 2 — Использование дополнительных функций

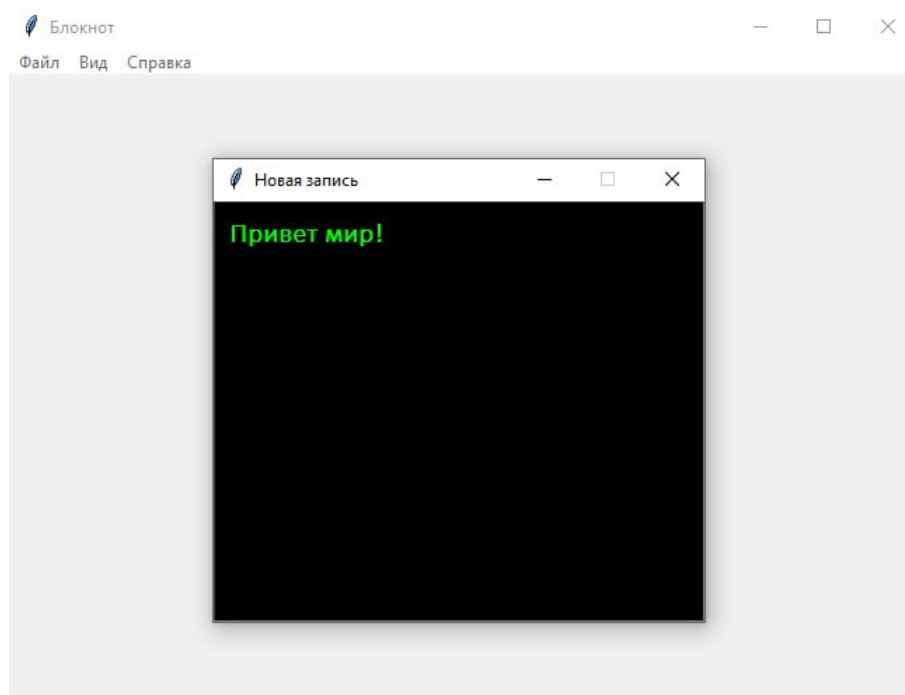


Рисунок 3 — Создание нового файла

Далее этап тестирования. На этом этапе давалось определение понятию тестирование, рассматривались его этапы и градации дефектов для составления тест-кейсов.

Программное обеспечение активно применяется пользователями и становится неотъемлемой частью их деятельности. Поэтому крайне важно обеспечить стабильность работы программного обеспечения на этапе разработки.

Ошибки в программах неизбежны. Поиск и выявление ошибок позволяет снизить финансовые и репутационные риски для разработчиков программы, найти и устранить критически ошибки в программном коде. Разработчикам и компаниям, занимающимся разработкой ПО, всегда следует стремиться найти все возможные ошибки до того, как программа поступит к заказчику или конечному потребителю.

Тестирование программного обеспечения (Software Testing) — проверка соответствия реальных и ожидаемых результатов поведения программы, проводимая на конечном наборе тестов, выбранном определённым образом.

Цель тестирования — проверка соответствия ПО предъявляемым требованиям, обеспечение уверенности в качестве ПО, поиск очевидных ошибок в

программном обеспечении, которые должны быть выявлены до того, как их обнаружат пользователи программы.

Для чего проводится тестирование ПО?

- Для проверки соответствия требованиям;
- Для обнаружения проблем на более ранних этапах разработки и предотвращения повышения стоимости продукта;
- Для обнаружения вариантов использования, которые не были предусмотрены при разработке;
- Для взгляда на продукт со стороны пользователя;
- Для повышения лояльности к компании и продукту, так как любой обнаруженный дефект негативно влияет на доверие пользователей.

Этапы тестирования:

- Анализ продукта;
- Работа с требованиями;
- Разработка стратегии тестирования и планирование процедур контроля качества;
- Создание тестовой документации;
- Тестирование прототипа;
- Основное тестирование;
- Стабилизация;
- Эксплуатация.

Стадии разработки ПО — этапы, которые проходят команды разработчиков ПО, прежде чем программа станет доступной для широкого круга пользователей.

Программный продукт проходит следующие стадии:

- анализ требований к проекту;
- проектирование;
- реализация;
- тестирование продукта;
- внедрение и поддержка.

Дефект (bug) — отклонение фактического результата от ожидаемого.

Серьёзность (severity) показывает степень ущерба, который наносится проекту существованием дефекта. Severity выставляется тестировщиком.

Градация серьёзности дефекта (Severity):

- Блокирующая;
- Критическая;
- Значительная;
- Незначительная;
- Тривиальная.

Срочность (priority) показывает, как быстро дефект должен быть устранён. Priority выставляется менеджером, тимлидом или заказчиком.

Градация приоритета дефекта (Priority):

- P1-Высокий. Критическая для проекта ошибка. Должна быть исправлена как можно быстрее.
- P2-Средний. Не критичная для проекта ошибка, однако требует обязательного решения.
- P3-Низкий. Наличие данной ошибки не является критичным и не требует срочного решения. Может быть исправлена, когда у команды появится время на ее устранение.

Тест-дизайн – это этап тестирования ПО, на котором проектируются и создаются тестовые случаи (тест-кейсы).

Приоритет и серьёзность багов – ключевые атрибуты, в соответствии с которыми определяется очередность исправления. Если неверно присвоить багу приоритет и серьёзность, эффективность исправления ошибки сильно снизится. Это может нанести вред бизнесу и привести к большим финансовым потерям.

В приложениях представлены исходные программные коды реализации.

Заключение. В работе были рассмотрены три важных этапа становления приложения: разработка, тестирование и ввод в эксплуатацию.

Стоит отметить, что процессы разработки и тестирования до того, как приложение запущено в эксплуатацию, могут длиться бесконечно долго, до тех пор, пока не будет определено, что дальнейшая разработка не требуется и не требуется дальнейшее тестирование, необходимые ошибки устранены. Однако чаще всего во время эксплуатации приложения пользователями выявляются ошибки, которые не были замечены во время тестирования. В таком

случае приходится останавливать работу приложения на время доработок, если ошибки критичны. В том случае, если ошибки не приводят к критичным последствиям, заказчик уведомляет разработчика об ошибках и данные ошибки устраняются. Таким образом, очевидно, что процесс проектирования приложения и его ввод в эксплуатацию представляет собой длительный и сложный процесс.

Созданная программа отличается достаточно простой формой, которая содержит все основные пункты технического задания, а именно создание, добавление и изменение записей в приложении. Программа обладает «мягким» и простым интерфейсом, доступным пользователю.

В результате проделанной работы были реализованы все поставленные задачи по созданию приложения, подобного современным программам, выполняющим функцию блокнотов, органайзеров, записных и телефонных книжек.