

Министерство цифрового развития, связи и массовых коммуникаций РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

“Сибирский Государственный Университет Телекоммуникаций и
Информатики”

(СибГУТИ)

Кафедра прикладной математики и кибернетики

Курсовая работа по дисциплине

"Визуальное программирование и человеко-машинное взаимодействие"

Тема: “Приложение записки”

Выполнила: студент 3 курса группы ИП-017

Пинчук А.А.

Проверил: доцент кафедры ПМиК

Мерзлякова Е.Ю.

Новосибирск, 2022

Содержание

| | |
|---|----|
| Часть 1 | 3 |
| Анализ задач и пользователей | 3 |
| Выбор репрезентативных задач | 4 |
| Заимствование..... | 6 |
| Черновое описание дизайна | 8 |
| Часть 2 | 13 |
| Описание функций разработанного приложения | 13 |
| Скриншоты интерфейса | 18 |
| Часть 3 | 27 |
| CWT-анализ | 27 |
| GOMS-анализ | 32 |
| Оценка интерфейса в соответствии с принципами Нильсена-Молиха.... | 40 |
| Принципы организации графического интерфейса | 41 |

Часть 1

Анализ задач и пользователей

Цель: найти двух человек, которые могут быть заинтересованы в решении предложенной задачи. Дайте их краткое описание (возраст, образование, профессия, навыки в выбранной сфере, навыки владения компьютером).

Решение: Приложением для создания «записок» будет интересно многим, кто много времени проводит за компьютером и ему зачастую нет времени чтобы лезть в блокнот или тетрадь, и ему будет удобно создавать информацию на экране. Поэтому, выбраны люди, которые будут много времени проводить за компьютером. Возраст необходимо брать как можно моложе, потому как с возрастом люди начинают реже сидеть за компьютером. Так же необходимо выбрать человека, который реже сидит за компьютером, но при этом ему было бы интересно данное приложение, чтобы можно было «сгладить» проблемы с приложением у других людей.

В качестве заинтересованных лиц выбраны следующие люди:

- 1) Однотруппник: возраст - 20 лет, образование - среднее общее, студент, навык владения компьютером - высокий.
- 2) Друг: возраст - 18 лет, образование - среднее общее, студент, навык владения компьютером - средний.

Выбор репрезентативных задач

Цель: перечислите репрезентативные задачи; затем все задачи, решение которых будет поддерживать разрабатываемая программа.

Решение:

Репрезентативные задачи:

1) Просмотр записей.

Программа должна предоставлять удобный интерфейс для просмотра записей.

2) Добавление записей.

Программа должна предоставлять удобный интерфейс для добавления записей.

3) Редактирование записей.

Программа должна предоставлять удобный интерфейс для редактирования записей.

4) Возможность сортировки записей.

Приложение должно предоставлять возможность сортировки записей.

5) Поиск записей

Приложение должно предоставлять возможность поиска записей.

6) Выставление напоминания

Приложение должно предоставлять возможность выставить напоминание для записи, а также должно уведомить, когда пришёл срок напоминания.

Второстепенные задачи:

1) Удаление записей.

Пользователь должен иметь возможность удалить запись.

2) Выделение важной информации в тексте записи

Приложение должно предоставлять возможность выделять важную информацию в тексте записи при помощи цвета и форматирования.

3) Выход из приложения.

Заимствование

Цель: найдите приложения или сайты, с которых можно заимствовать какие-либо решения интерфейса, приведите ссылку на источник. Эти приложения не обязательно должны выполнять точно такие же задачи. Заимствовать можно что угодно, даже расположение кнопок. Выберите и напишите, что именно Вы будете заимствовать из данных приложений и зачем.

Решение:

1) Sticky notes.

Из приложения позаимствовано расположение кнопки для создания новой записки и кнопка закрытия записки. Так же позаимствовано расположение строки поиска.

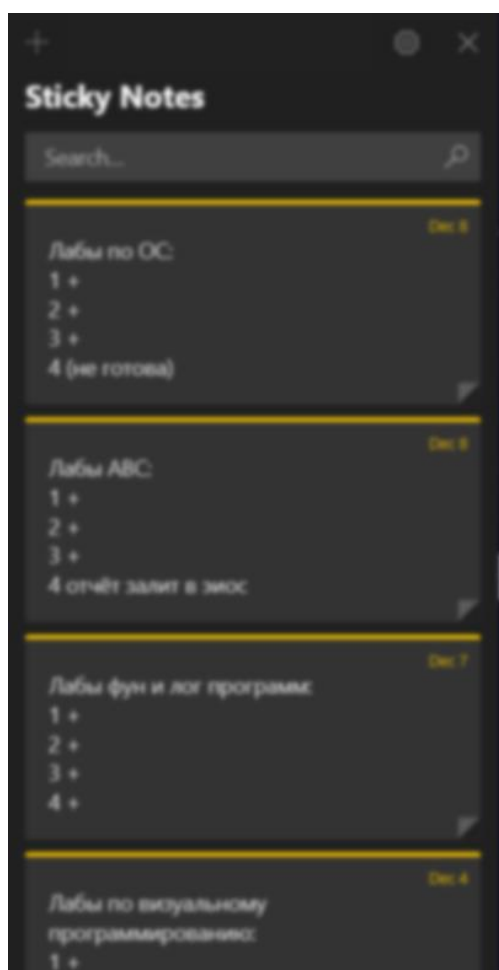


Рисунок 1. Приложение Sticky notes

2) Microsoft To Do.

Из данного приложения позаимствовано интерфейсное решение задания даты напоминания.



Рисунок 2. Приложение Microsoft To Do

3) Microsoft Word.

Из данного приложения я взял элементы редактирования и их расположение для редактирования записки.

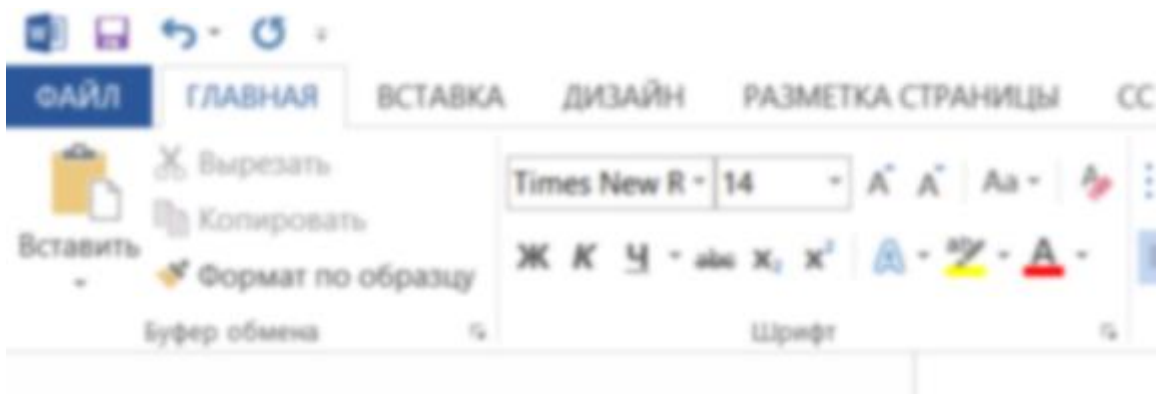


Рисунок 3. Приложение Microsoft To Do

Черновое описание дизайна

Цель: опишите черновой вариант дизайна словами и графически (иллюстрации с пояснениями). Черновой вариант должен отражать все внешние элементы интерфейса и их назначение.

Решение:

Размеры и положения элементов интерфейса относительно и могут меняться в процессе создания.

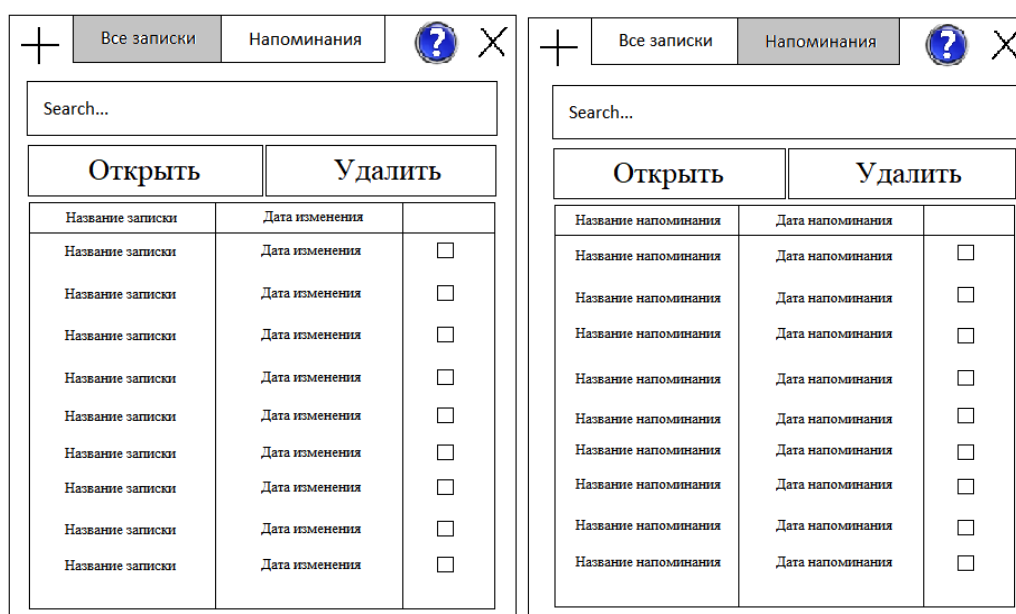


Рисунок 4. Черновой вариант главного окна приложения

Главное окно будет содержать:

- 1) Объект “TabWidget” для переключения между вкладками;
- 2) Объект "Button” для добавления записи.

На каждой вкладке будет содержаться:

- 1) Объект “LineEdit” для осуществления поиска среди записей;
- 2) 2 объекта “Button” для открытия и удаления записи;
- 3) Объект “TableWidget” для отображения доступных записей;
- 4) В 3 колонке “TableWidget” будет объект “CheckBox” для выбора записей для открытия/закрытия.

Название записки

Название записки

Текст записки

Рисунок 5. Черновой вариант окна записки

Окно записки будет содержать:

- 1) Объект “LineEdit” для названия записки недоступный для редактирования;
- 2) Объект “PlainTextEdit” для текста записки недоступный для редактирования;

При нажатии на любое поле записки записка перейдёт в режим редактирования:

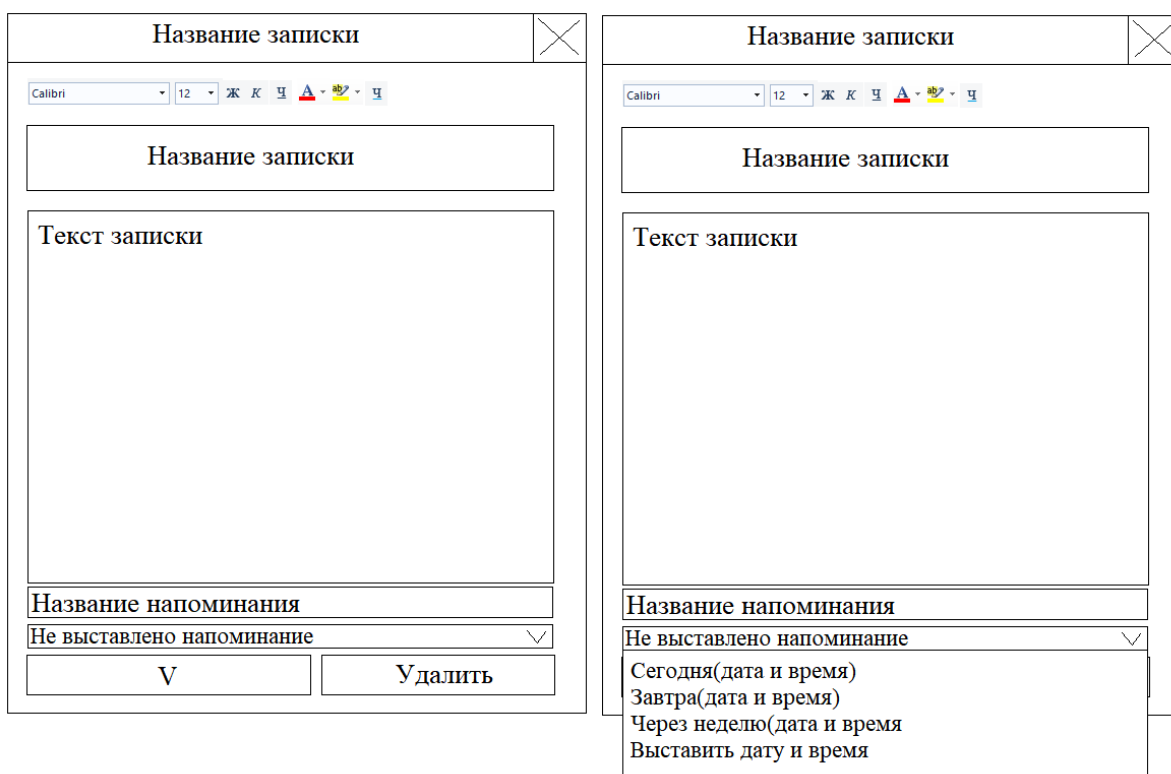


Рисунок 6. Черновой вариант окна записки в режиме редактирования

Окно записки в режиме редактирования будет содержать:

- 1) 2 объекта “LineEdit” для названия записки доступный для редактирования и для названия напоминания;
- 2) Объект “PlainTextEdit” для текста записки доступный для редактирования;
- 3) Объект “FontComboBox” для задания стиля шрифта;
- 4) 2 объекта “ComboBox” для задания размера шрифта и для задания даты и времени напоминания;
- 5) 6 объектов “PushButton” для редактирования текста, а именно: задания жирного шрифта, курсива, подчёркнутого текста, цвета фона, текста и подчёркивания;
- 6) 2 объекта “PushButton” для сохранения изменений и скрытия интерфейса редактирования и удаления данной записки.

При создании записи все поля должны быть заполнены. Если одно из полей не заполнено, то кнопка появится предупреждение заполнить все формы.

Если закрыть окно в режиме редактирования, то приложение предложит сохранить записку. Если закрыть окно в обычном режиме, то приложение попросит подтверждение.

При нажатии на кнопки задания цвета текста, фона и подчёркивания откроется стандартное окно ColorDialog, а при нажатии на «Выставить дату и время» в “ComboBox” для даты и времени напоминания, откроется форма выставления даты и времени:

The image shows a sketch of a software window titled "Напоминание" (Reminder). The window contains a "Календарь" (Calendar) section, which is currently empty. Below the calendar is a time input field showing "0:00" with up and down arrow buttons. At the bottom left is a green checkmark button, and at the bottom right is a red X button.

Рисунок 7. Черновой вариант окна выставления времени

Окно выставления времени будет содержать:

- 1) 2 объекта “PushButton” для сохранения времени и отмены изменений;
- 2) Объект “CalendarWidget” для выставления даты напоминания;
- 3) Объект “TimeEdit” для выставления времени напоминания.

Часть 2

Описание функций разработанного приложения

В программе реализовано 6 классов:

1) Главное окно;

on_pushButton_2_clicked() – создаёт новую запись задаёт ей минимальный размер, добавляет её ко всем записям, сообщает классу редактора что запись новая и соединяет все сигналы редактора со слотами главного окна.

on_pushButton_clicked() – открывает старые записи или если они открыты, то возвращает их из свёрнутого состояния и показывает поверх других окон.

NewNote(NotesInform inf) – добавляет новую запись, отображает её в tablewidget и посылает сигнал обратно в редактор, который передаёт редактору его номер среди записей(для возможности редактирования).

EditedNote(NotesInform inf) – получает информацию при окончании редактирования записи, находит эту запись среди номеров открытых записей и по этому номеру записывает информацию в массив и отображает изменённую информацию.

CloseNote(int NoteId) – получает номер закрытой записи и удаляет её из номеров открытых.

DeleteNote(int NoteId) – получает номер удалённой записи и удаляет её из главного окна и базы данных.

on_lineEdit_textChanged(const QString &arg1) – получает строку поиска, находит совпадения среди названий записок и выводит результат поиска.

on_lineEdit_2_textChanged(const QString &arg1) – получает строку поиска, находит совпадения среди названий напоминаний и выводит результат поиска.

on_pushButton_4_clicked() – открывает старые записи или если они открыты, то возвращает их из свёрнутого состояния и показывает поверх других окон.

on_pushButton_5_clicked() – удаляет выделенные в таблице записок записи.

on_pushButton_6_clicked() – удаляет выделенные в таблице напоминаний записи.

on_pushButton_3_clicked() – открывает окно справки.

timerEvent() – вычисляет текущую дату и время, сверяет её с напоминаниями и в случае совпадения открывает или показывает поверх других окон записку, формируя при этом сообщение с текстом напоминания.

QVector<NotesInform> getInformationDB() – берёт информацию из базы данных.

showNotes(QVector<NotesInform> &inf) – отображает информацию на главном окне.

deleteAllItems(QTableWidget* wid) – удаляет все элементы из таблицы.

QVariantList fillVariantList(NotesInform inform) – заполняет список для отправления информации на запись в базу данных.

closeEvent(QCloseEvent *event) – событие закрытия окна.

WriteInBase() – запись информации в базу данных.

getLastNoteId() – передаёт номер для новой записки.

WriteCoordinates() – записывает координаты открытых окон для дальнейшей записи в базу данных.

OpenEdit(QTableWidget* tab) – открывает старые записи или если они открыты, то возвращает их из свёрнутого состояния и показывает поверх других окон.

DeleteEdit(QTableWidget* tab) – получает номер удалённой записи и удаляет её из главного окна и базы данных.

ResearchLabel(QLabel* lab, QString arg1) – получает строку поиска, находит совпадения среди названий записок или напоминаний и выводит результат поиска.

NoteIdReceive(int NoteId) – сигнал который формируется для передачи нового номера записи.

2) Окно записки;

on_pushButton_7_clicked() – формирует сигналы сохранения информации в записке или создания новой записи.

on_pushButton_8_clicked() – формирует сигнал удаления записки из базы данных и закрывает окно.

on_comboBox_2_activated(int index) – определяет что пользователь хочет сам задать время напоминания и создаёт форму с датой и временем.

DateReceived(QDateTime date) – получает данные из формы с датой и временем.

NoteIdReceived(int NoteId) – получает номер и записывает его в свои данные для дальнейшего редактирования.

Edit_button_clicked() – показывает скрытые элементы редактирования.

on_pushButton_clicked() – подчёркивает выделенный текст или если он уже подчёркнут, то убирает подчёркивание.

on_pushButton_2_clicked() – задаёт цвет фона выделенного текста.

on_pushButton_3_clicked() – задаёт цвет подчёркивания выделенного текста и если текст не подчёркнут – подчёркивает его.

on_fontComboBox_currentFontChanged(const QFont &f) – задаёт шрифт выделенного текста.

on_pushButton_4_clicked() – делает курсивом выделенный текст или если он уже курсивом, то делает обычным.

on_comboBox_activated(const QString &arg1) – задаёт размер шрифта выделенного текста.

on_pushButton_5_clicked() – делает жирным выделенный текст и если он уже жирный, то делает обычным.

fillInf() – берёт информацию из всех форм.

closeEvent(QCloseEvent *event) – проверяет что все формы заполнены, и в случае режима редактирования спрашивает у пользователя сохранить данные или в случае обычного режима спрашивает подтверждение у пользователя на закрытие записки.

ShowEdit() – показывает элементы редактирования

HideEdit() – прячет элементы редактирования.

RememberMessage() – выводит текст напоминания.

on_pushButton_9_clicked() – в случае если записка новая, то окно закрывается; в случае если записка старая, то закрывается режим редактирования без сохранения изменений.

3) Окно задания даты напоминания:

on_pushButton_clicked() – формирует всё в одну строку и посылает дату в редактор.

on_pushButton_2_clicked() – закрывает окно.

4) Окно справки:

changedUrl(QUrl url) – получает url нажатый в html, берёт название файла и открывает его.

on_pushButton_clicked() – открывает главное окно справки.

on_pushButton_3_clicked() – открывает предыдущее окно справки.

on_pushButton_2_clicked() – открывает следующее окно справки.

Скриншоты интерфейса

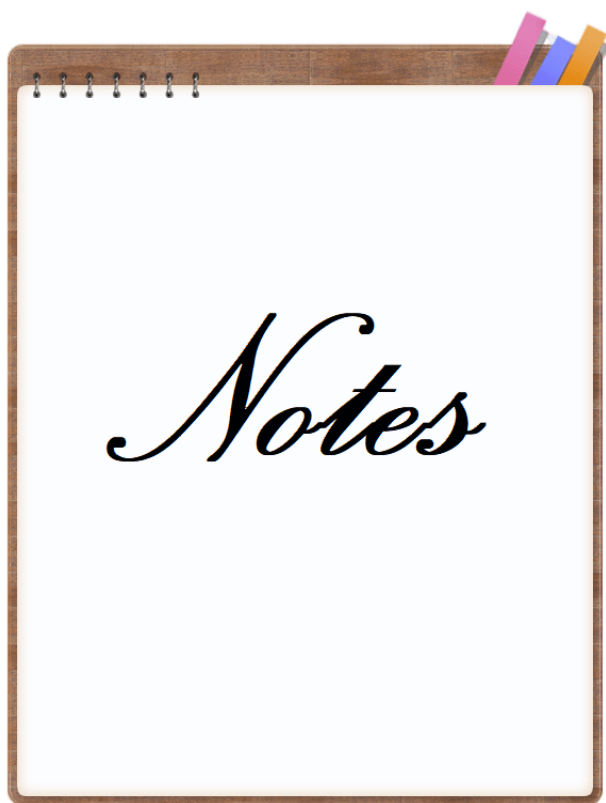


Рисунок 8. Загрузочное окно.

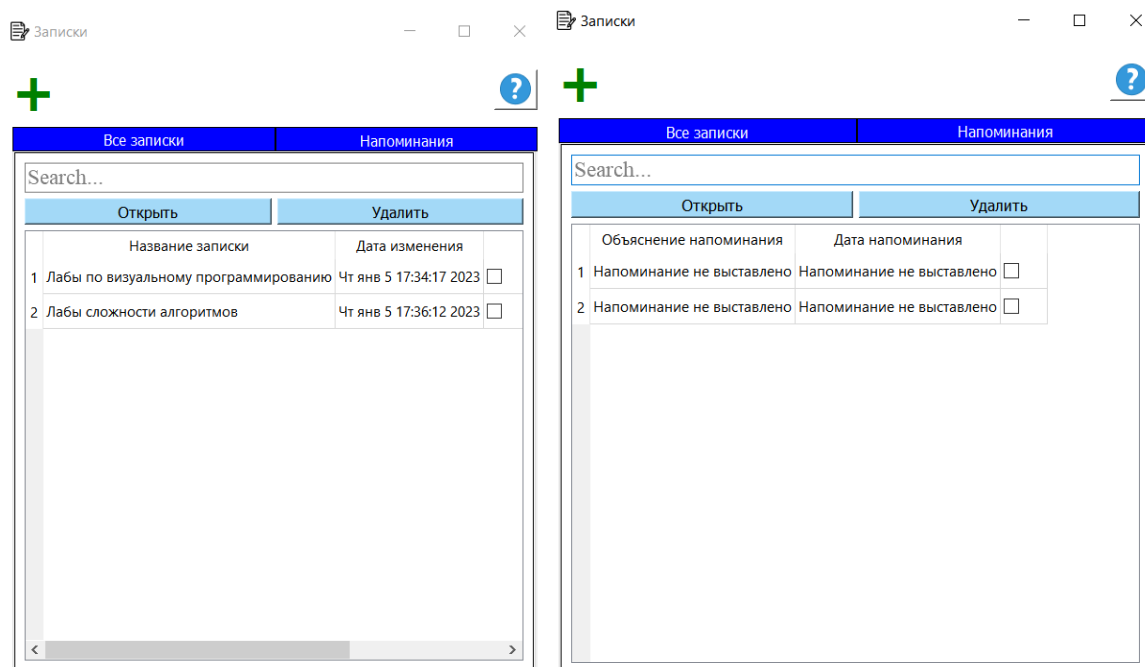


Рисунок 9. Главное окно приложения.

Form

Название записки

Название записки

Редактирование текста

Times New Roman 14

ж к ч а А ч

Текст записки

Напоминание

Объяснение напоминания

Нет напоминания

Сохранить Удалить

Отменить

Рисунок 10. Окно новой записки

Если нажать на кнопку “Сохранить” с пустыми полями, то выведет информационное окно о том, что нужно заполнить все поля.

The image shows a software window titled "Form" with standard Windows window controls (minimize, maximize, close). The window contains several input fields and buttons:

- Название записки**: A text input field containing the placeholder text "Название записки".
- Редактирование текста**: A section containing a font family dropdown menu set to "Times New Roman" and a font size dropdown menu set to "14". Below these are six small icons for text formatting: bold (ж), italic (к), underline (ч), bulleted list (ay), strikethrough (A), and text color (Ч).
- Текст записки**: A large text area for the note's content.
- Напоминание**: A section with a text input field containing "Объяснение напоминания" and a dropdown menu set to "Нет напоминания".
- Buttons**: Three large blue buttons at the bottom labeled "Сохранить", "Удалить", and "Отменить".

An error dialog box is overlaid on the "Текст записки" area. The dialog has a title bar with a close button (X) and the text "Ошибка". The main text of the dialog says "Заполните все формы". There is a single "ОК" button at the bottom of the dialog.

Рисунок 11. Информационное окно

При нажатии на кнопку “Удалить” выводится окно для подтверждения удаления.

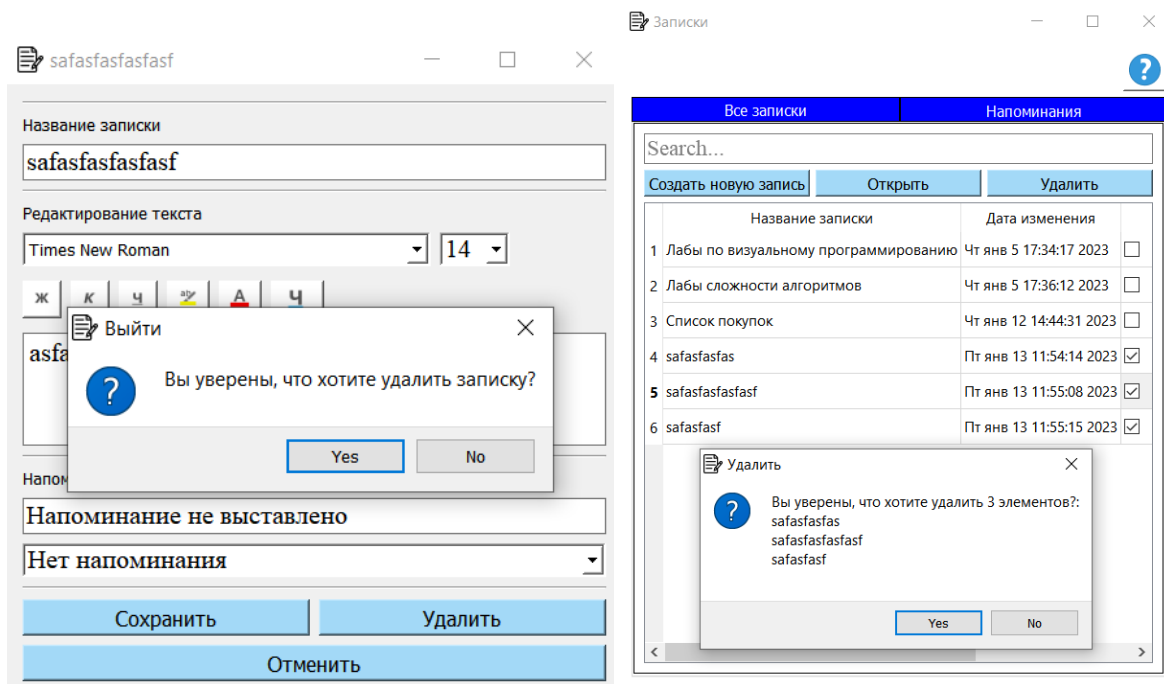


Рисунок 12. Окно подтверждения удаления.

При выборе в нижнем ComboBox «Выбрать дату и время» откроется форма, где можно выставить дату и время.

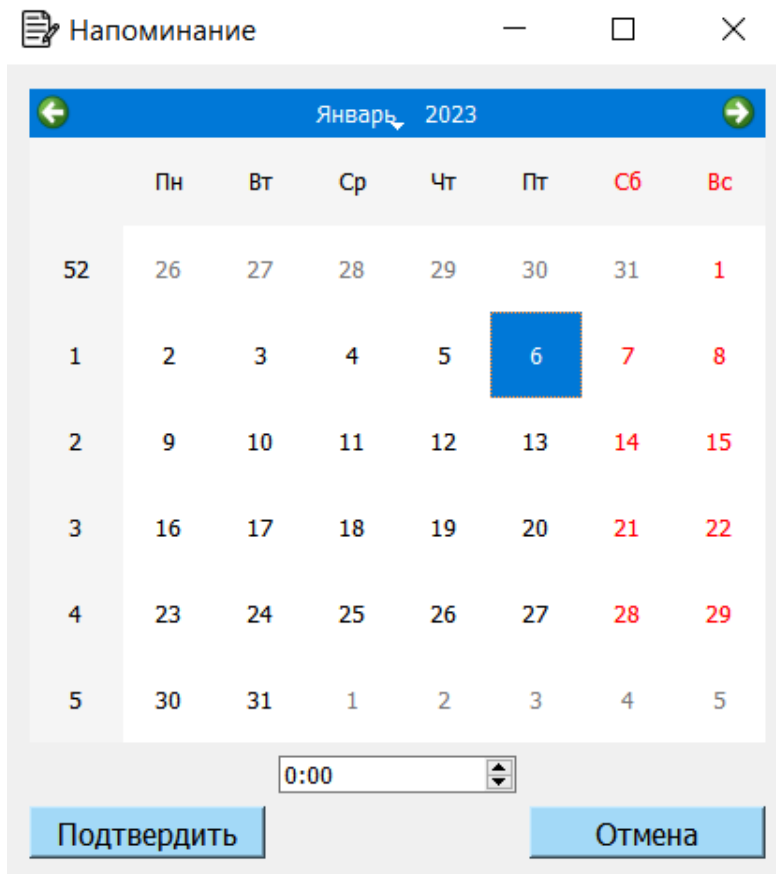


Рисунок 13. Окно даты и времени.

Окно записки без режима редактирования:

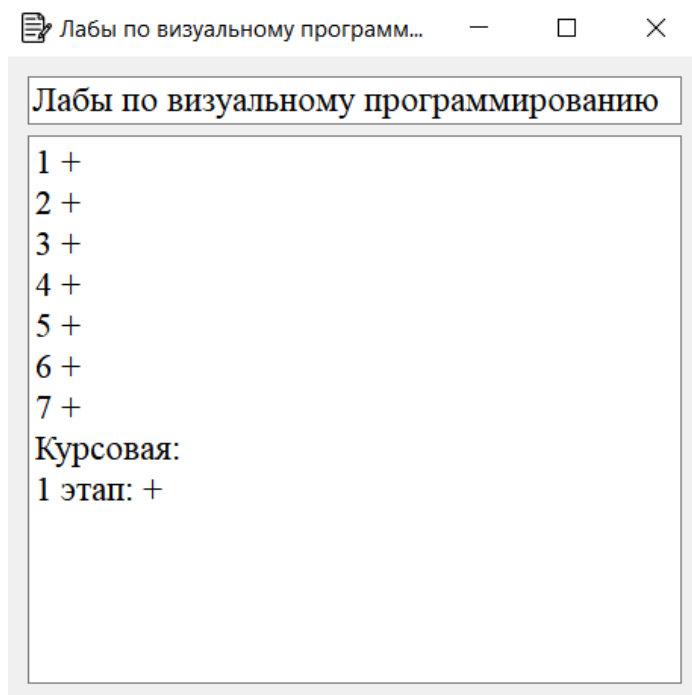


Рисунок 14. Окно записки

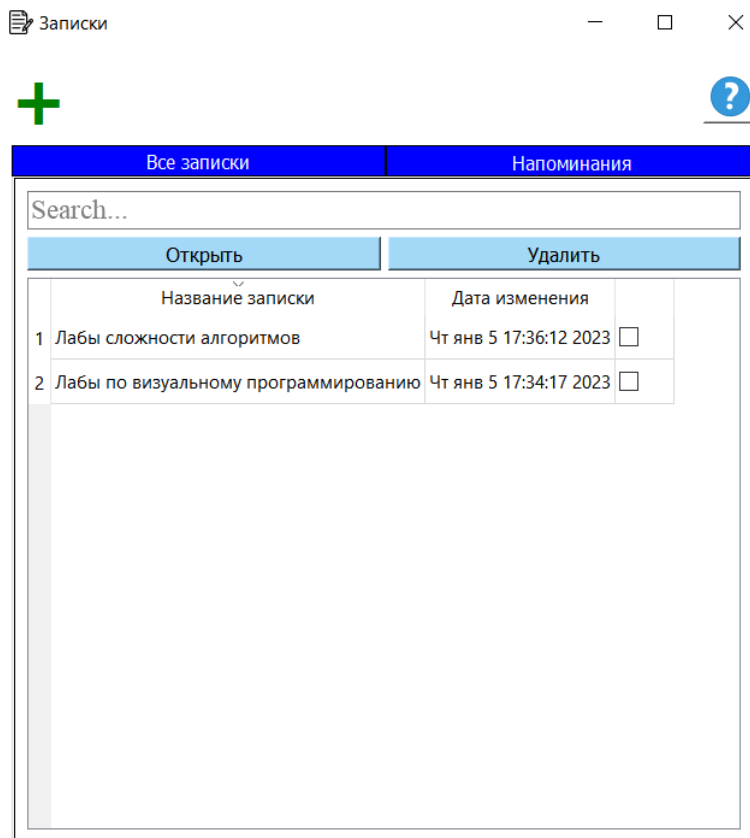


Рисунок 15. Сортировка таблицы по названию записки

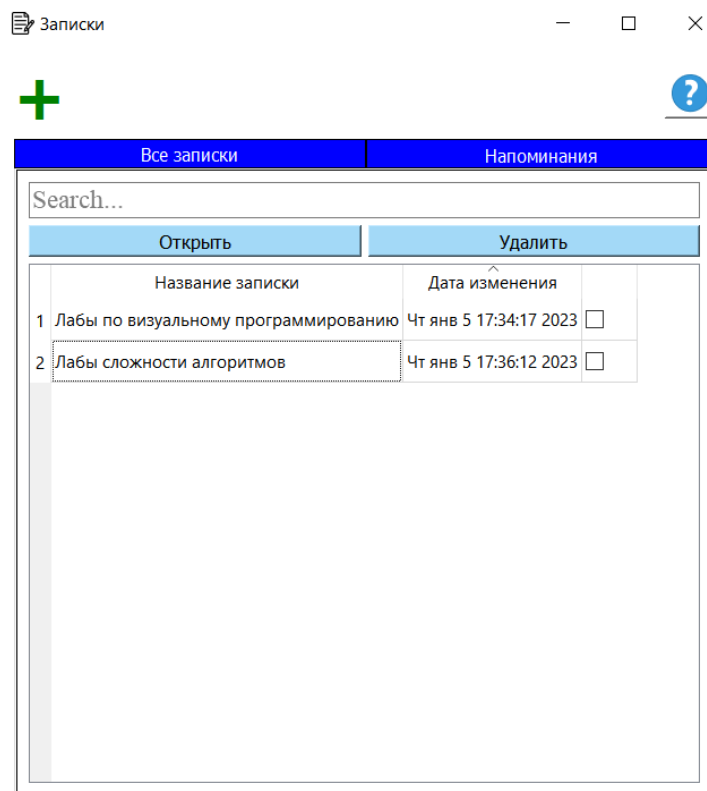


Рисунок 16. Сортировка таблицы по названию записки

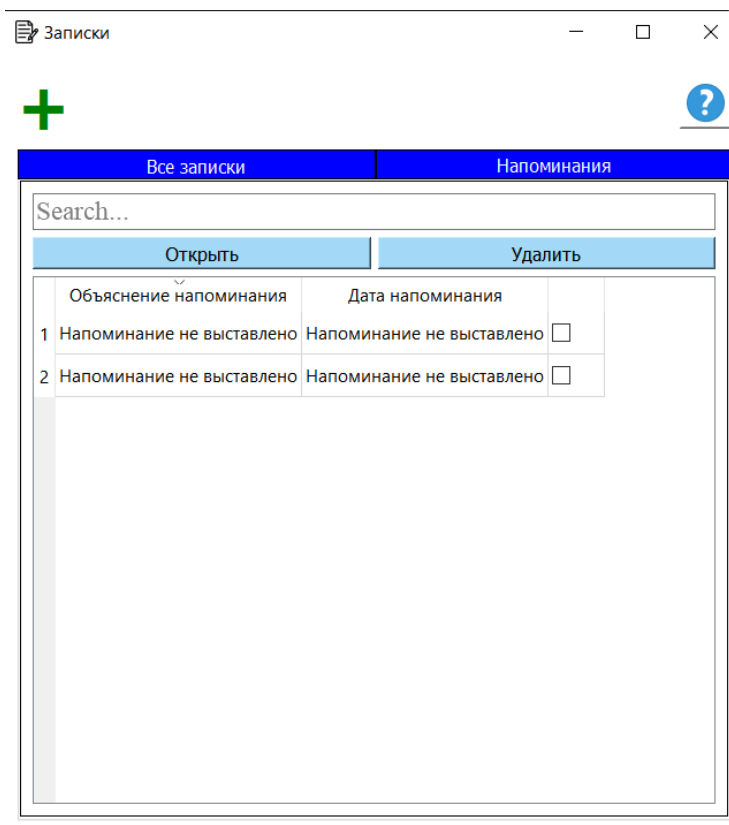


Рисунок 17. Сортировка напоминаний по названию напоминания

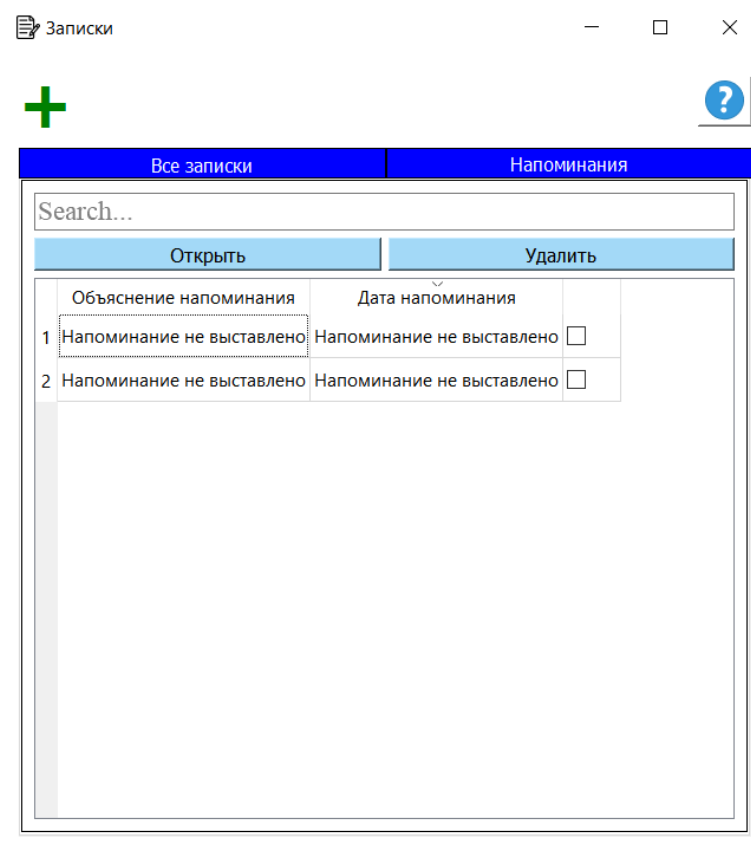


Рисунок 18. Сортировка напоминаний по дате напоминания

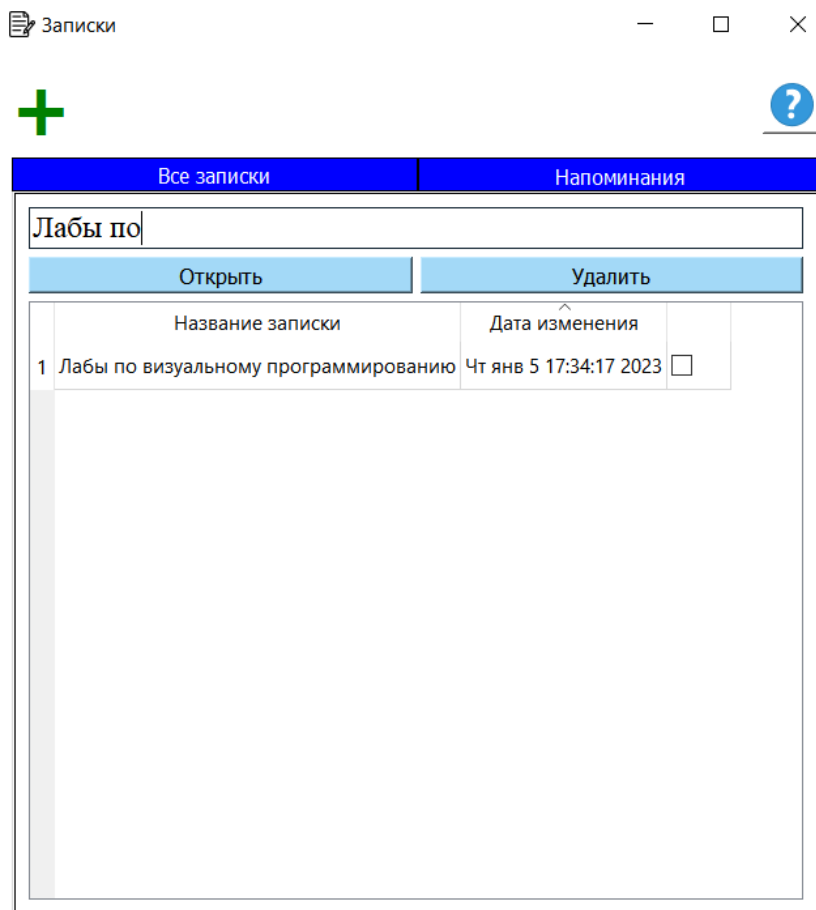


Рисунок 19. Поиск по названию записки

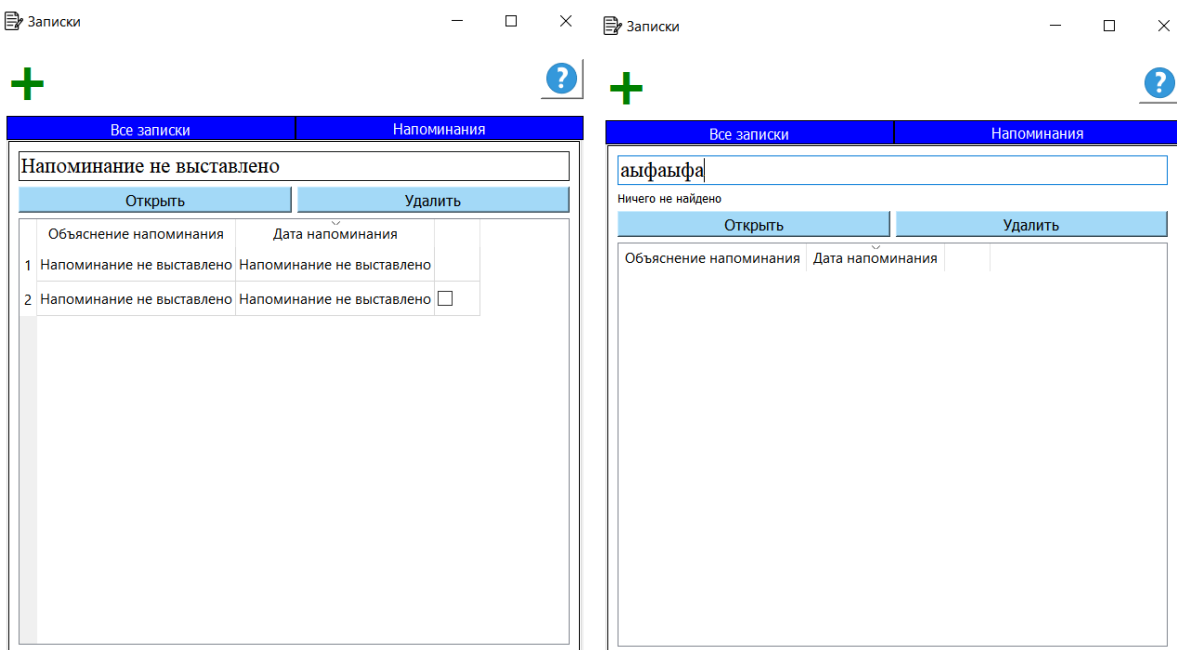


Рисунок 20. Поиск по названию напоминания

Приложение записки

Аннотация

Это приложение предназначено для создания записок и предоставления возможности их форматирования и использования.

Как пользоваться приложением:

- [Как создать новую запись](#)
- [Как удалить запись](#)
- [Как редактировать запись](#)
- [Как найти нужную запись или напоминание](#)
- [Как отсортировать записки](#)

[Назад](#)[На главную](#)[Вперёд](#)

Рисунок 21. Окно справки

Часть 3

CWT-анализ

Выберем 2 задачи, которые поддерживает интерфейс приложения, затем определим список действий, выполняемых пользователем.

1. Удаление записки из главного окна.
2. Создать новую записку, отредактировать, закрыть, открыть и расположить на экране записку.

CWT-анализ первой части:

1. Выбор записи. Пользователю необходимо нажать на checkbox в таблице на тех строках, которые хочешь удалить.
2. Удаление. После выбора записи следует нажать кнопку “Удалить”. Выводится окно для подтверждения удаления, в нём следует подтвердить или отменить удаление.

Найденные проблемы:

Для данной задачи интерфейс может вызвать непонимание, т. к. только нажав на кнопку “Удалить” пользователь будет ждать, когда он сможет выбрать строку для удаления, но в ответ ничего не произойдет.

Возможные способы решения:

Можно сделать кнопку не активной пока пользователь не выберет строки. Либо при нажатии на кнопку выводить окно с предупреждением, что необходимо выбрать строки.

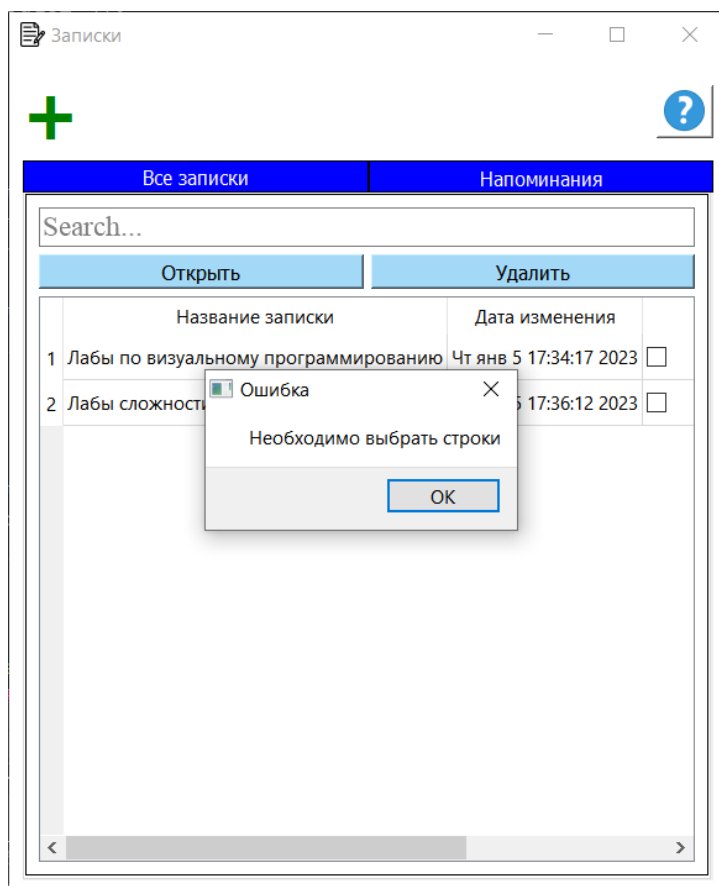


Рисунок 22. Исправленная проблема удаления

CWT-анализ второй части:

1. Создание новой записи. Пользователю необходимо кликнуть на кнопку «+».
2. Редактирование. Пользователю необходимо заполнить все поля открытой формы и выставить напоминание.
3. Закреть. Пользователю необходимо нажать на кнопку кнопку «Сохранить», а потом кнопку «Закреть» и подтвердить закрытие приложения, либо сразу нажать кнопку «Закреть» и пользователю предложат сохранить записку.
4. Открыть. Пользователю необходимо выбрать записку, которую он только что создал, поставить напротив неё галочку и нажать кнопку «Открыть»

5. Расположить на экране. Пользователю необходимо расположить записку на экране и растянуть записку так, как ему будет удобно.

Найденные проблемы:

1. Для пользователя не совсем очевидно, что большой зелёный плюс может значить создание новой записи, а поэтому он будет искать на экране и скорее нажмёт на кнопку «Открыть» вместо того чтобы нажать большой зелёный плюс.

2. В редакторе кроме очевидных из Microsoft Word иконок редактирования появляется кнопка с цветом подчёркивания, что неочевидно для пользователя.

Возможные способы решения:

1. Создать высвечивающуюся подсказку при наведении на кнопку «+» и предупреждение на кнопку «Открыть» с текстом «Если хотите создать новую записку – нажмите на «+»», либо объединить функции кнопки «+» и кнопки «Открыть», и изменить текст кнопки на «Открыть/Создать новый», что значительно упростит интерфейс и все нужные пользователю для работы элементы будут собраны в одном месте, но при этом станет неочевидным, что чтобы открыть старую запись нужно поставить галочку напротив нужной записи, поэтому лучше всего убрать кнопку «+» и добавить рядом с кнопками «Открыть» и «Удалить» кнопку «Создать новую запись».

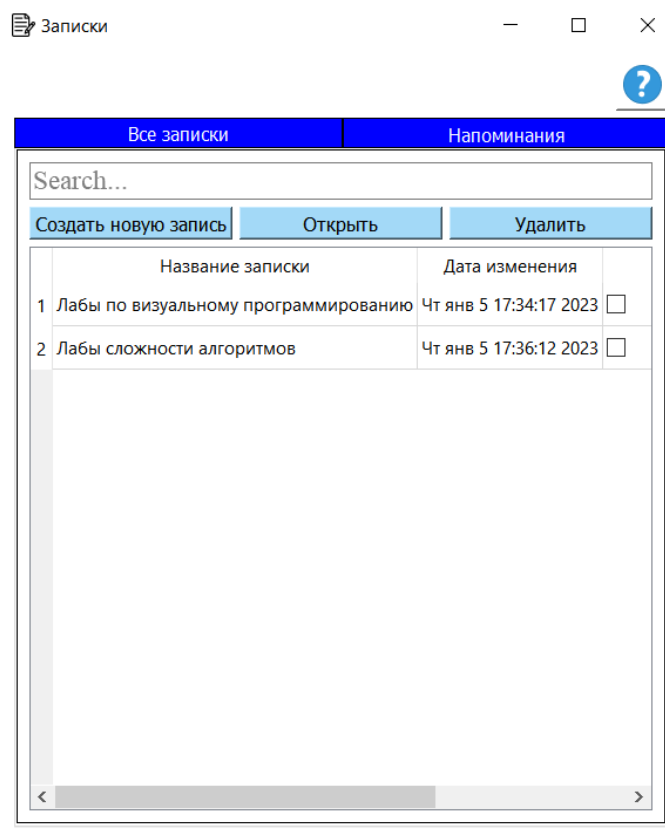


Рисунок 23. Исправленная проблема создания новой записи

2. Добавить всплывающую подсказку при наведении на все элементы интерфейса редактирования.




Лабы по визуальному программированию

Название записки

Лабы по визуальному программированию

Редактирование текста

Times New Roman 14

ж к ч   

Цвет подчёркивания

1 +
2 +
3 +
4 +
5 +
6 +
7 +
Курсовая:
1 этап: +

Напоминание

Напоминание не выставлено

Пн янв 16 05:00:00 2023

Сохранить Удалить

Отменить

Рисунок 24. Исправленная проблема редактирования текста

GOMS-анализ

Цель 1: Создать новую записку, отредактировать, закрыть, открыть и расположить на экране записку.

Для выполнения цели сформулируем подцели:

1. Создать новую записку и отредактировать её.
2. Сохранить и закрыть.
3. Открыть.
4. Расположить окно на экране и задать нужный размер.

Теперь опишем методы для каждой подцели:

1. Создать новую записку и отредактировать её.
 - 1.1. Кликнуть на кнопку «Создать новую запись».
 - 1.2. Заполнить поля формы и задать напоминание.

Для конкретности результатов зададим конкретные данные, которыми нужно заполнить записку:

Список покупок

Название записки

Список покупок

Редактирование текста

Times New Roman 14

ж к ч

Хлеб
Молоко
Яйца
Колбаса
Масло
Сахара

Напоминание

Иди в магазин

Сохранить Удалить

Отменить

Рисунок 25. Пример заполненной формы

Будем считать, что перед каждым новым элементом «Списка покупок» пользователь не думает, а точно знает, что ему нужно напечатать. Так же в напоминании нужно выставить завтрашнюю дату 10 утра.

2. Сохранить и закрыть.

2.1. Нажать кнопку «Сохранить».

2.2. Нажать кнопку «Закрыть».

2.3. Подтвердить закрытие нажав на кнопку «Yes».

3. Открыть.

3.1. Ставим галочку напротив последней записки (которую мы только что создали).

3.2. Нажимаем кнопку «Открыть».

4. Расположить окно на экране и задать нужный размер.

4.1. Перетаскиваем окно в нужное место.

4.2. Задаёт нужный размер окна при помощи мыши.

Теперь распишем каждый метод с точностью до операций:

1.1 Кликнуть на кнопку «Создать новую запись».

М (мыслительная подготовка)

Н (перенос руки на мышь)

Р (переместить указатель мыши)

В (клик мыши)

1.2. Заполнить поля формы и задать напоминание.

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

Н (перенос руки на клавиатуру)

14 К (нажатие клавиши)

Н (перенос руки на мышь)

Р (переместить указатель мыши)

В (клик мыши)

Н (перенос руки на клавиатуру)

37 К (нажатие клавиши)

Н (перенос руки на мышь)

Р (переместить указатель мыши)

В (клик мыши)

Н (перенос руки на клавиатуру)

13 К (нажатие клавиши)

Н (перенос руки на мышь)

Р (переместить указатель мыши)

В (клик мыши)

Р (переместить указатель мыши)

В (клик мыши)

Р (переместить указатель мыши)

В (клик мыши)

Р (переместить указатель мыши)

2 В (клик мыши)

Н (перенос руки на клавиатуру)

2 К (нажатие клавиши)

Н (перенос руки на мышь)

Р (переместить указатель мыши)

В (клик мыши)

2.1. Нажать кнопку «Сохранить».

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

2.2. Нажать кнопку «Заккрыть».

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

2.3. Подтвердить закрытие.

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

3.1. Ставим галочку напротив последней записки (которую мы только что создали).

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

3.2. Нажимаем кнопку «Открыть».

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

4.1. Перетаскиваем окно в нужное место.

М (мыслительная подготовка)

Р (переместить указатель мыши)

Д (проведение с помощью мыши прямой линии)

4.2. Задаёт нужный размер окна при помощи мыши.

М (мыслительная подготовка)

Р (переместить указатель мыши)

Д (проведение с помощью мыши прямой линии)

Расчёт:

1.1. МНРВ

1.2. МРВН 14К НРВН 37К Н РВН 13К НРВРВРВР 2В Н 2К НРВ

2.1. МРВ

2.2. МРВ

2.3. МРВ

3.1. МРВ

3.2. МРВ

4.1. МРД

4.2. МРД

Получаем:

$$\text{МНРВ} = 1,35 + 0,4 + 1,1 + 0,2 = 3,05 \text{ с}$$

$$\begin{aligned} \text{МРВН 14К НРВН 37К Н РВН 13К Н 3РВ Р 2В Н 2К НРВ} &= 1,35 + 1,1 + 0,2 \\ &+ 0,4 + 14 * 0,2 + 0,4 + 1,1 + 0,2 + 0,4 + 37 * 0,2 + 0,4 + 1,1 + 0,2 + 0,4 + 13 * \\ &0,2 + 0,4 + 3 * (1,1 + 0,2) + 1,1 + 2 * 0,2 + 0,4 + 2 * 0,2 + 0,4 + 1,1 + 0,2 \\ &= 27,95 \text{ с} \end{aligned}$$

$$\text{МРВ} = 1,35 + 1,1 + 0,2 = 2,65 \text{ с}$$

$$\text{МРД} = 1,35 + 1,1 + 2 = 4,45 \text{ с}$$

$$T = 3,05 + 27,95 + 2,65 * 5 + 4,45 * 2 = 53,15 \text{ с}$$

Итог: 53,15 с

Цель 2: Удаление записки из главного окна.

Для выполнения цели сформулируем подцели:

1. Выбрать записку.
2. Нажать удалить.
3. Подтвердить удаление.

Теперь опишем методы для каждой подцели:

1. Выбрать записку.

1.1. Поставить галочку напротив записки которую нужно удалить.

2. Нажать удалить.

2.1. Нажать на кнопку «Удалить»

3. Подтвердить удаление.

2.1. Нажать на кнопку «Yes»

Теперь распишем каждый метод с точностью до операций:

- 1.1. Поставить галочку напротив записки которую нужно удалить.

М (мыслительная подготовка)

Н (перенос руки на мышь)

Р (переместить указатель мыши)

В (клик мыши)

- 1.2. Нажать на кнопку “Удалить”.

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

2.1. Нажать кнопку “Yes”.

М (мыслительная подготовка)

Р (переместить указатель мыши)

В (клик мыши)

Расчёт:

МНРВМРВМРВ

Расставляем операторы М и получаем:

$$\text{МНРВМРВМРВ} = 1,35 + 0,4 + 1,1 + 0,2 + 1,35 + 1,1 + 0,2 + 1,35 + 1,1 + 0,2 = 8,35 \text{ с}$$

Итог: 8,35 с

Оценка интерфейса в соответствии с принципами Нильсена-Молиха

1.Простой и естественный диалог

В интерфейсе отсутствует не относящейся к теме или редко используемая информация. Информация, которая выводится на экран появляется в естественном порядке, соответствующем ожиданиям пользователя.

2.Говорите на языке пользователя

В программе используются слова и понятия из мира пользователя. Не используются специфические инженерные термины.

3.Минимизируйте загрузку памяти пользователя

Вся нужная для пользователя информация выводится только тогда, когда он выбирает какой-либо пункт меню.

4.Будьте последовательны

У пользователя имеется возможность изучить действия в одной части программы и применить их снова, чтобы получить похожие результаты в других местах.

5.Обеспечьте обратную связь

Пользователь получает обратную связь о совершенных действиях. Например, после добавления записи в таблицу выводится окно о успешном добавлении и сразу отображается новая запись в таблице.

6.Обеспечьте хорошо обозначенные выходы

Если пользователь попадает в часть системы, которая его не интересует, у него всегда есть возможность выйти оттуда. В каждом открывшемся окне есть кнопка “Отменить”, которая закроет его.

7. Хорошие сведения об ошибках.

При введении данных в формы добавления или редактирования, программа сообщает об отсутствии информации в полях ввода.

8. Предотвращайте ошибки.

Пользователь не сможет выбрать или записать неверную информацию в пункт «ID клиента», «Статус заказа».

9. Снабдите программу системой помощи.

В главном окне приложения находится пункт меню «Помощь», где описаны все принципы работы программы. Для открытия навигатора нужно нажать на кнопку «Помощь» → «Справка».

Принципы организации графического интерфейса

Принцип кластеризации. Все кнопки для конкретной функции группируются вместе, также они выделены цветом.

Принцип “видимость отражает полезность”. Часто используемые элементы расположены на видном месте и являются легкодоступными.

Принцип интеллектуальной последовательности. Используются похожие экраны для похожих функций. Например, окно редактирования записи и окно добавления записи.

Принцип “цвет как приложение”. Цвета в приложении использованы умеренно. Только нужные части интерфейса выделены цветом.

Принцип уменьшения беспорядка. На экране приложения ничего лишнего не находится. Видны только часто используемые элементы, использован минимум цвета.

Вывод

В процессе реализации курсового проекта были пройдены все этапы создания продукта, рассмотренные в курсе “Визуальное программирование и человеко-машинное взаимодействие”. Были поставлены задачи для реализации и описан черновой вариант будущего приложения. Приложение реализовано в соответствии с поставленными задачами.