Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

Уфимский университет науки и технологий

Институт информатики, математики и робототехники

Кафедра вычислительной математики и кибернетики

Отчет к лабораторной работе №3

по дисциплине «Проектирование человеко-машинного интерфейса»

по теме «Разработка проекта прототипа пользовательского интерфейса»

Выполнили: студенты группы ПРО-331Б

Семенов Л.А.

Семенов Г.Д.

Проверила:

Нагимова И. А.

Уфа 2024

**Цель**: изучение особенностей построения прототипов пользовательского интерфейса и принципов их стандартизации в контексте решаемой задачи, сформулированной при изучении и анализе действий пользователя в рамках предпроектного обследования.

**Вариант задания**:

Для выполнения лабораторной работы был выбран вариант *магазин запчастей для автомобиля.*

**Задачи**:

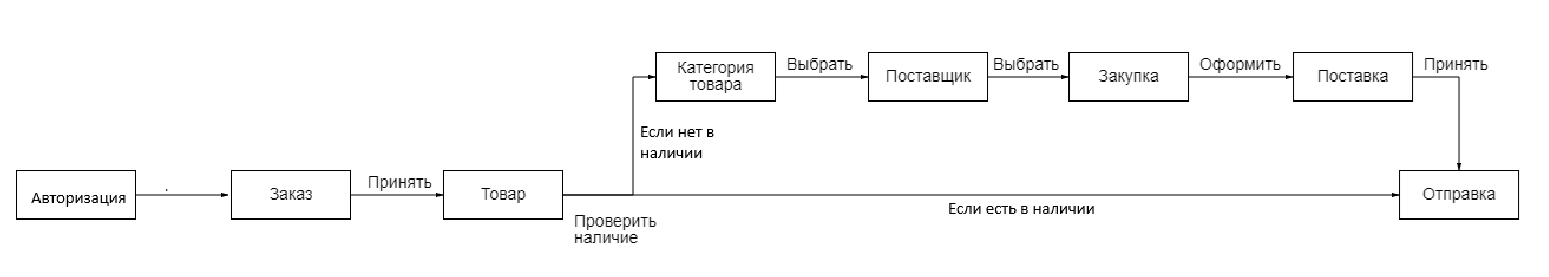
В рамках лабораторной работы решаются следующие задачи: приобретение навыков проектирования прототипов пользовательских интерфейсов и обоснования выбора управляющих элементов с учетом возможной взаимосвязи объектов и субъектов интерфейса.

**Ход работы**:

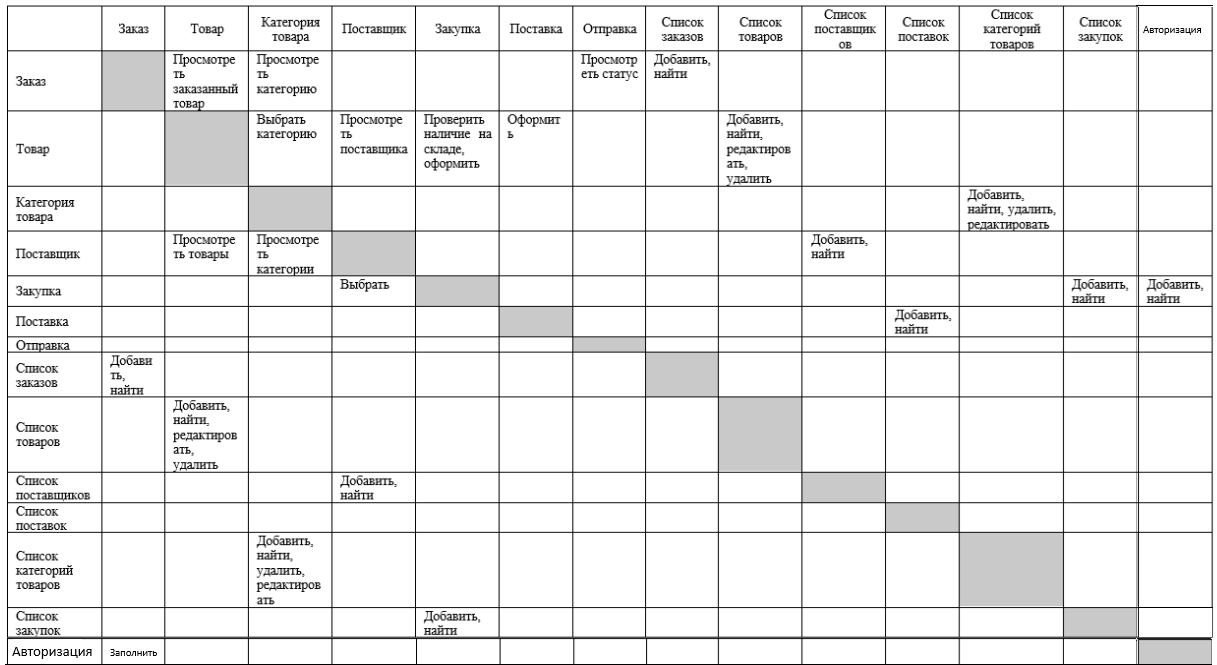
1. Таблица объектов и действий:

|  |  |  |
| --- | --- | --- |
| Объекты и данные | Тип объекта | Действия |
| Заказ | Активный | Заполнить |
| Поиск товара | Активный | Проверить наличие |
| Поставщик | Активный | Выбрать |
| Закупка | Активный | Оформить |
| Поставка | Пассивный | Проверить статус |
| Категория товара | Активный | Выбрать |
| Отправка | Пассивный | Проверить статус |
| Список заказов | Активный | Найти, добавить |
| Список товаров | Активный | Найти, добавить, удалить, редактировать |
| Список поставщиков | Активный | Найти, добавить |
| Список поставок | Активный | Найти, добавить |
| Список категорий товаров | Активный | Найти, добавить, удалить, редактировать |
| Список закупок | Активный | Найти, добавить |
| Авторизация | Активный | Заполнить |

1. Схема взаимодействия объектов:



1. Матрица прямого манипулирования объектами:



1. Перечисление элементов интерфейса, их функциональной нагрузки и обоснование их выбора:

|  |  |  |
| --- | --- | --- |
| **Элемент интерфейса** | **Функциональная нагрузка** | **Обоснование их выбора** |
| Заказ | Элемент dataGridView | **DataGridView** обеспечивает удобный просмотр и управление заказами, позволяя легко сортировать и фильтровать данные. |
| Товар | Элемент dataGridView, button | **DataGridView** используется для отображения списка товаров, а **button** - для добавления новых товаров и управления ими. |
| Поставщик | Элемент dataGridView | **DataGridView** позволяет эффективно организовать и редактировать информацию о поставщиках. |
| Закупка | Элемент dataGridView, button | **DataGridView** отображает детали закупок, а **button** используется для подтверждения или отмены закупок. |
| Поставка | Элемент dataGridView | **DataGridView** идеально подходит для отслеживания статуса и деталей поставок. |
| Категория товара | Элемент dataGridView, listBox | **DataGridView** для просмотра категорий, а **listBox** для удобного выбора из списка. |
| Отправка | Элемент dataGridView | **DataGridView** обеспечивает четкое представление о графиках и статусах отправок. |
| Список заказов | dataGridView | **DataGridView** позволяет управлять и анализировать заказы в одном месте. |
| Список товаров | dataGridView | **DataGridView** облегчает инвентаризацию и управление запасами товаров. |
| Список поставщиков | dataGridView | **DataGridView** предоставляет централизованный доступ к информации о поставщиках. |
| Список поставок | dataGridView | **DataGridView** помогает координировать и отслеживать поставки. |
| Список категорий товаров | dataGridView | **DataGridView** упрощает классификацию и поиск товаров по категориям. |
| Список закупок | dataGridView | **DataGridView** используется для мониторинга и управления процессом закупок. |
| Авторизация | button | **button** - для ввода пароля. |

* Элемент **dataGridView** используется для отображения и управления списками данных. Он позволяет пользователям просматривать, сортировать и редактировать данные в табличном виде, что делает его идеальным для работы со списками заказов, товаров, поставщиков и т.д.
* Элемент **button** применяется для выполнения команд или действий, таких как добавление нового товара или заказа, подтверждение закупки и т.д. Кнопки предоставляют простой способ взаимодействия с программой, выполняя определенные функции, связанные с данными.
* Элемент **listBox** используется для отображения списка элементов, из которых пользователь может выбрать один или несколько. В контексте категорий товаров, это позволяет легко просматривать и выбирать различные категории.

Выбор этих элементов интерфейса обоснован их способностью эффективно управлять и представлять данные в удобной и понятной форме, что является ключевым для любой системы управления базами данных.

1. Примерный интерфейс:



Рисунок 1. Форма со списком заказов:

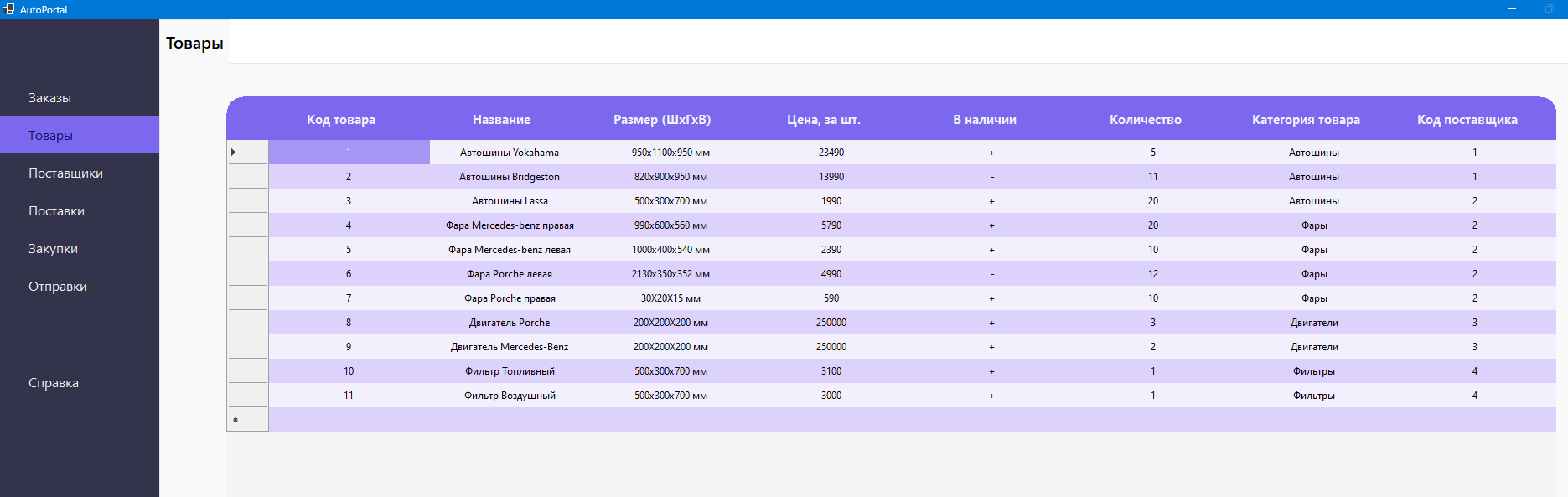


Рисунок 2. Форма со списком товаров

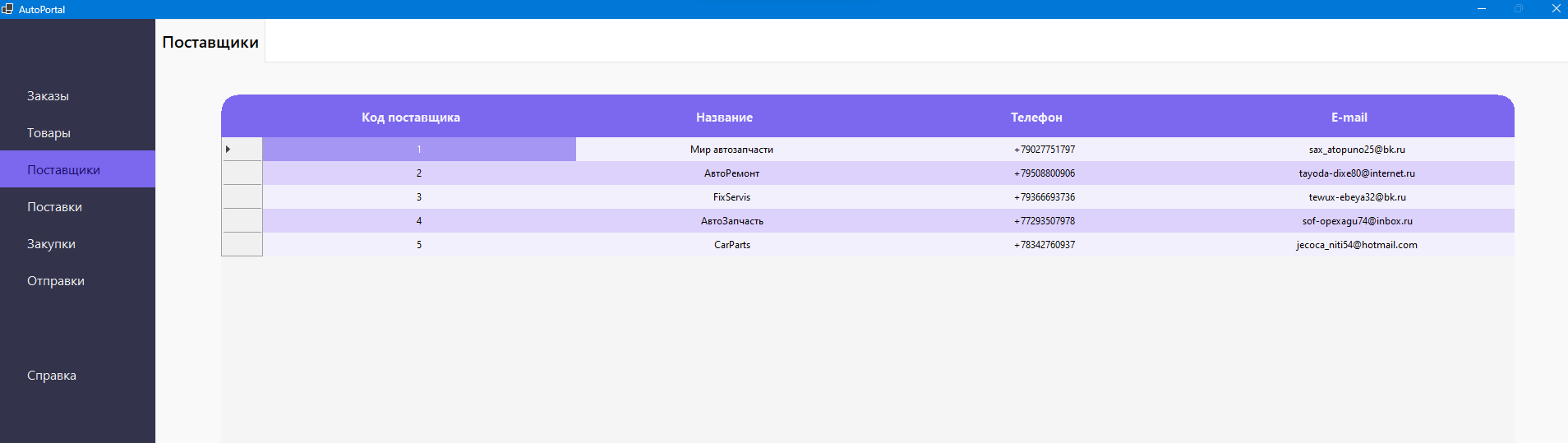


Рисунок 3. Форма со списком поставщиков

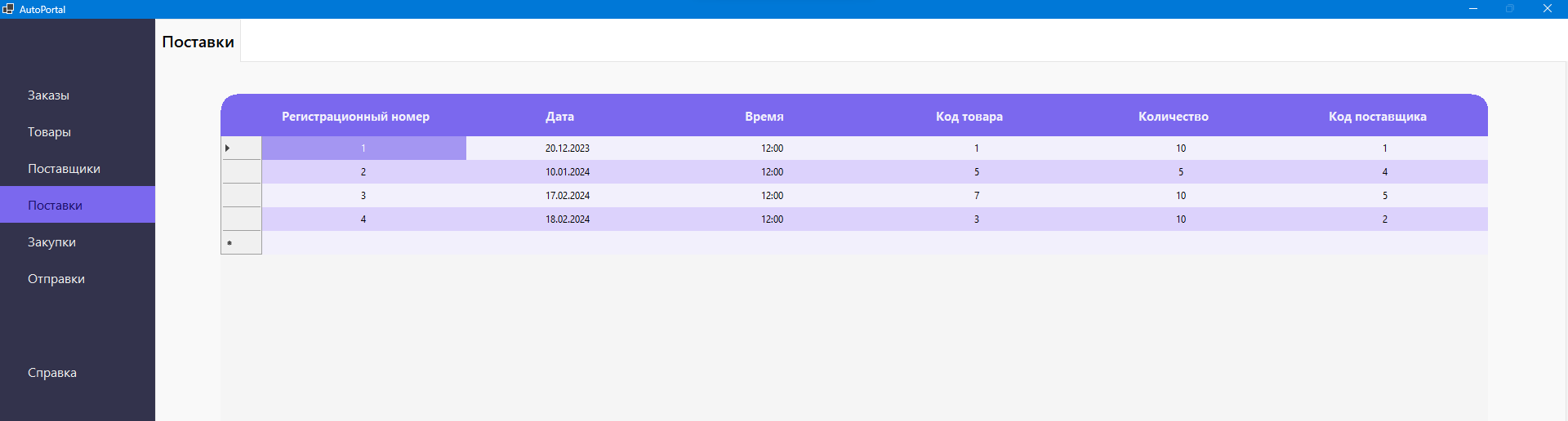


Рисунок 4. Форма со списком поставок



Рисунок 5. Формы с другими списками выглядят аналогично.



Рисунок 6. Форма для оформления закупок товаров.

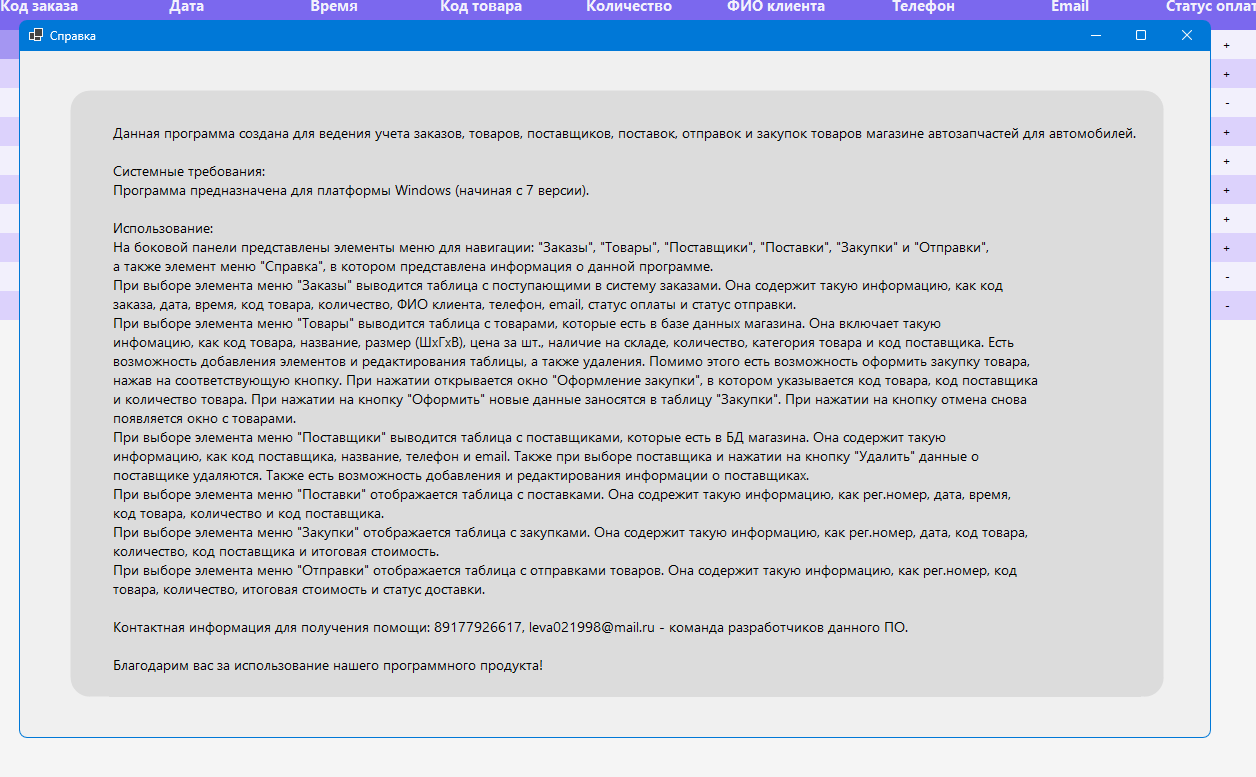


Рисунок 7. Справочная информация.

**Вывод:**

В рамках лабораторной работы были приобретены навыки разработки прототипов пользовательского интерфейса и принципов их стандартизации в контексте решаемой задачи, сформулированной при изучении и анализе действий пользователя в рамках предпроектного обследования.

**Контрольные вопросы:**

1. **Перечислите основные принципы проектирования пользовательского интерфейса**.

При построении интерфейса важно придерживаться определенных принципов, способствующих обеспечению его унификации и дружественности:

– контроль пользователем интерфейса. У пользователя создается субъективное ощущение управления системой, делающей его взаимодействие с системой более комфортным;

– уменьшение загрузки памяти пользователя. Элементы, редко или вовсе не используемые пользователем должны располагаться «на заднем плане» или быть скрыты от пользователя, уступая место наиболее актуальным, часто используемым. Элементам управления;

– последовательность пользовательского интерфейса. Все элементы управления в своей совокупности должны «читаться»: у пользователя не должно возникать сомнения, в каком порядке ему следует использовать те или иные элементы.

1. **Какие типы объектов вы знаете?**

Активные и пассивные.

1. **Какие типы данных циркулируют между объектами? Как они выражаются?**

В программировании различают разные типы данных, которые могут циркулировать между объектами:

1. Целочисленные данные (integers)

2. Дробные числа (floats)

3. Строки (strings)

4. Логические значения (booleans)

5. Списки (lists)

6. Словари (dictionaries)

7. Кортежи (tuples)

8. Наборы (sets)

9. Объекты (objects)

10. Файлы (files)

1. **Что представляет собой матрица прямого манипулирования?**

Эта матрица представляет собой двумерную таблицу, строки и столбцы которой описывают исходные и конечные объекты взаимодействия экземпляров классов, выявленных на предыдущих этапах. Для того, чтобы описать все возможные варианты взаимодействия, необходимо и в столбцах, и в строках таблицы указать все выявленные на первом этапе проектирования объекты и данные, независимо от того, к какому виду они относятся (активные или пассивные). На пересечении строк и столбцов таблицы должны быть заданы связывающие их действия.