

АНОО "Физтех-лицей" им. П. Л. Капицы

Отчёт по практикуму "Оптовый склад"

Выполнили:

Титов Фёдор 11И

Севрюков Никита 11И

Содержание

1	Основные задачи	2
2	Диаграмма классов	2
3	Спецификация классов	2
4	Диаграмма объектов	3
5	Модульная (файловая) структура программы	4
6	Используемый инструментарий	5
7	Пользовательский интерфейс	5

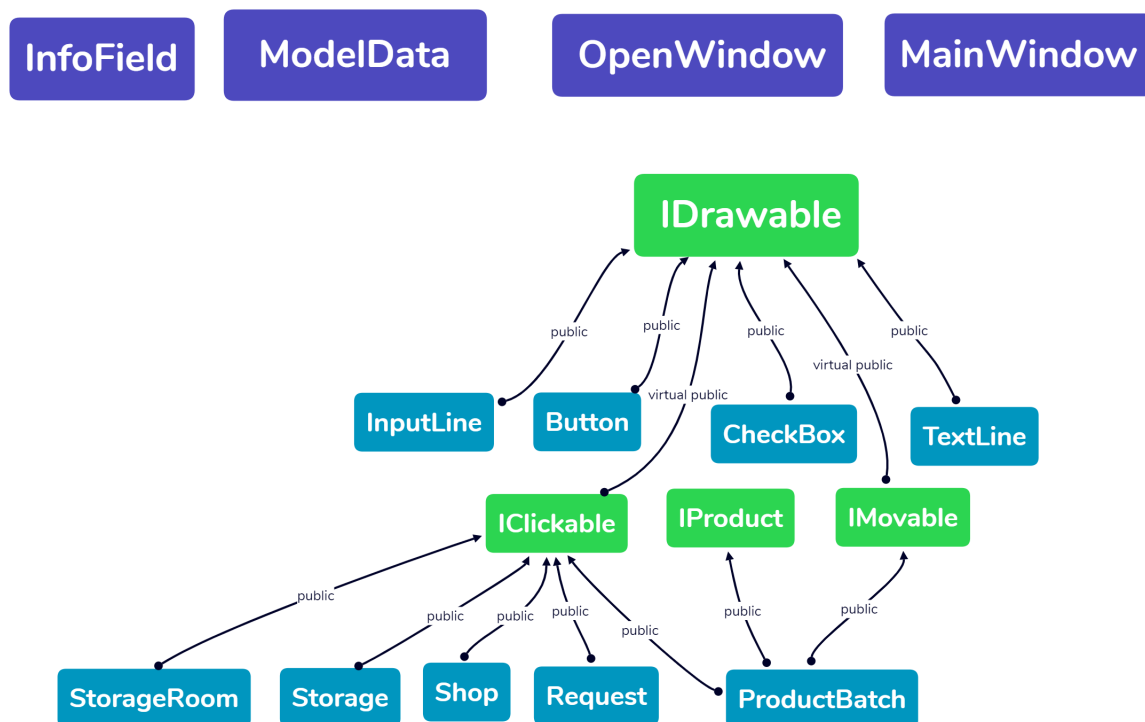
1 Основные задачи

Главная задача: составить программу, осуществляющую имитационное моделирование процесса, а также визуализирующую этот процесс.

Цель: необходимо создать имитацию (математическую модель) работы оптового склада, обслуживающего М торговых точек в течение N дней, на котором хранятся К видов товаров.

Разработана программа с использованием математической модели, составляющей *выгодный для магазинов* заказ тех или иных продуктов с учётом вероятности, а также имитирующая поведение грузоперевозки и хранения различных товаров.

2 Диаграмма классов



3 Спецификация классов

Описаны основные (в т. ч. интерфейсы) классы программы. Остальные классы отвечают за визуальную составляющую.

```
class IDrawable // все предметы, что могут быть нарисованы
class IProduct // информация о конкретном продукте
class IClickable : virtual public IDrawable // все предметы, что могут быть нажаты
class IMoveable : virtual public IDrawable // все предметы, что могут двигаться

class ModelData // класс, хранящий в себе информацию о мат. модели:
// цены, кол-во дней, магазинов, начальное кол-во товара на складе и т. д.
```

```

class Storage : public IClickable // основной класс, отвечающий за всю информацию
// на складе и операции с товарами

class StorageRoom : public IClickable // хранит информацию о "комнате" с одним
// и тем же видом товара

class ProductBatch : public IClickable, public IProduct, public IMovable
// партия продуктов - это какое-то количество оптовых упаковок одного и того же вида
// товара, произведённого в одну и ту же дату (например, 15 оптовых упаковок
// молока, произведённого 3 дня назад). Это позволяет наглядно
// отобразить срок годности товаров и обрабатывать их.

class Shop : public IClickable // определённый магазин, содержит метод
// эффективного выбора товара

```

4 Диаграмма объектов

```

class Shop {
    StorageRoom* rooms_[15]; // информация о каждой "комнате" с товаром
}

class StorageRoom {
    std::deque<ProductBatch*>batches_; // хранимые на данный момент партии
    // определённого товара
    std::vector<ProductBatch*>purchases_; // заказанные партии товара в
    // текущий момент с поставщиков
}

class Request {
    // содержит в себе всю информацию о заказе от определённого магазина
    int profit = 0; // выгода с покупки товара
    int product_type_; // тип товара
    int product_count_; // кол-во оптовых упаковок
    std::wstring name_; // название магазина
    Shop* customer_; // указатель на магазин
}

class Shop {
    Request* MakeRequest(storage* Storage) // функция заказа от магазина на склад
    // использует методы класса Storage
}

class ModelData {
    int number_days_; // кол-во дней
    int number_shops_; // кол-во магазинов
    int number_products_ = 0; // кол-во используемых продуктов
}

```

```

    bool used_products_[15]; // использованные товары
    int count_products_[15]; // и их кол-во на складе
}

```

5 Модульная (файловая) структура программы

- Data (вспомогательные файлы) $\rightarrow \begin{cases} CORBEL.TTF \\ ArialRegular.ttf \\ AAVANTEBS.TTF \end{cases}$
- SFML (файлы библиотеки) $\rightarrow \begin{cases} doc \\ examples \\ include \\ lib \end{cases}$
- Frontend.h/Frontend.cpp (классы) $\rightarrow \begin{cases} IDrawable \\ Button \\ CheckBox \\ InputLine \\ IMovable \\ TextLine \end{cases}$
- InfoField.h/Infofield.cpp (класс) \rightarrow InfoField
- MainWindow.h/MainWindow.cpp (класс) \rightarrow MainWindow
- ModelData.h/ModelData.cpp (класс) \rightarrow ModelData
- OpenWindow.h/OpenWindow.cpp (класс) \rightarrow OpenWindow
- Products.h/Products.cpp (классы) $\rightarrow \begin{cases} IProduct \\ IClickable \\ ProductBatch \\ StorageRoom \\ Storage \end{cases}$
- Shop.h/Shop.cpp (классы) $\rightarrow \begin{cases} Request \\ Shop \end{cases}$
- main.cpp

6 Используемый инструментарий

Библиотеки:

1. SFML – для работы с визуальной частью программы (*Frontend*)
2. Стандартные библиотеки STL: `iostream`, `vector`, `algorithm`, `deque`, `random`, `cmath` - для работы с реализующей частью программы (*Backend*)

Для работы с вероятностью был использован генератор случайных чисел `std::mt19937`.

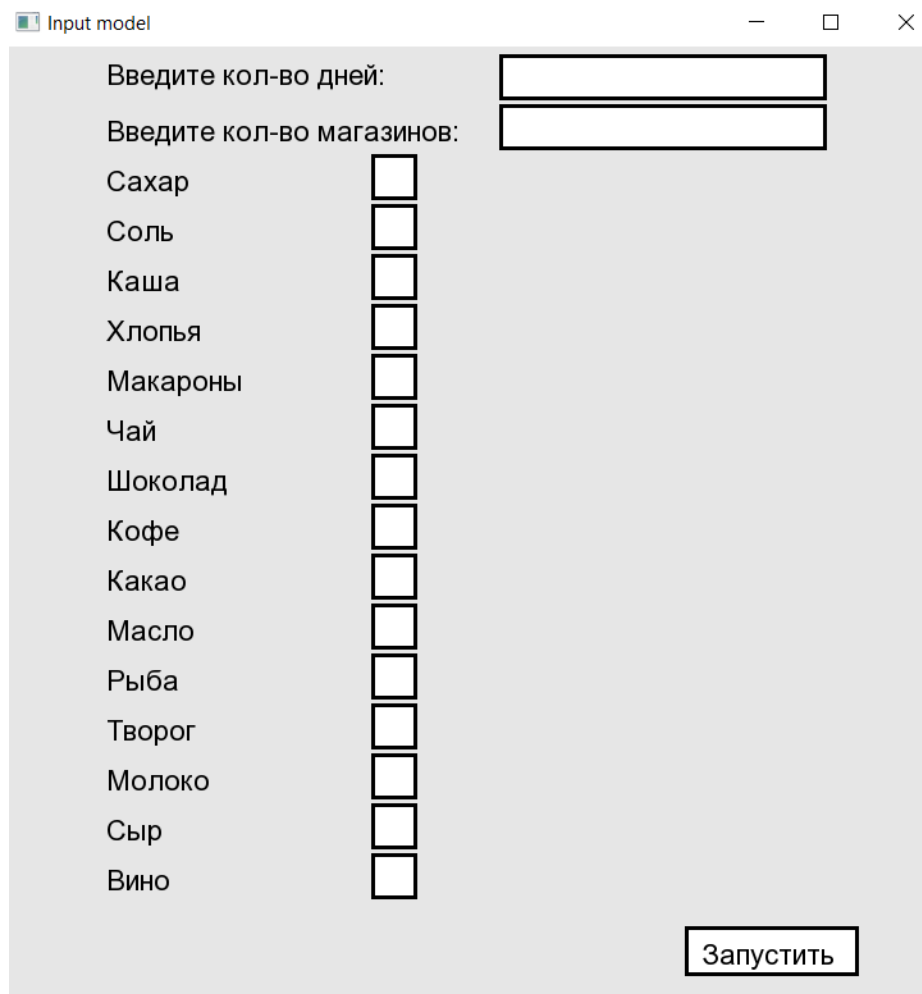
7 Пользовательский интерфейс

После запуска пользователь встречает окно для заполнения модели. В нём он может ввести кол-во дней и магазинов, а также отметить товары, которые **уже есть** на складе и их кол-во оптовых упаковок.

Максимально допустимое кол-во дней – 30

Максимальное кол-во магазинов – 7

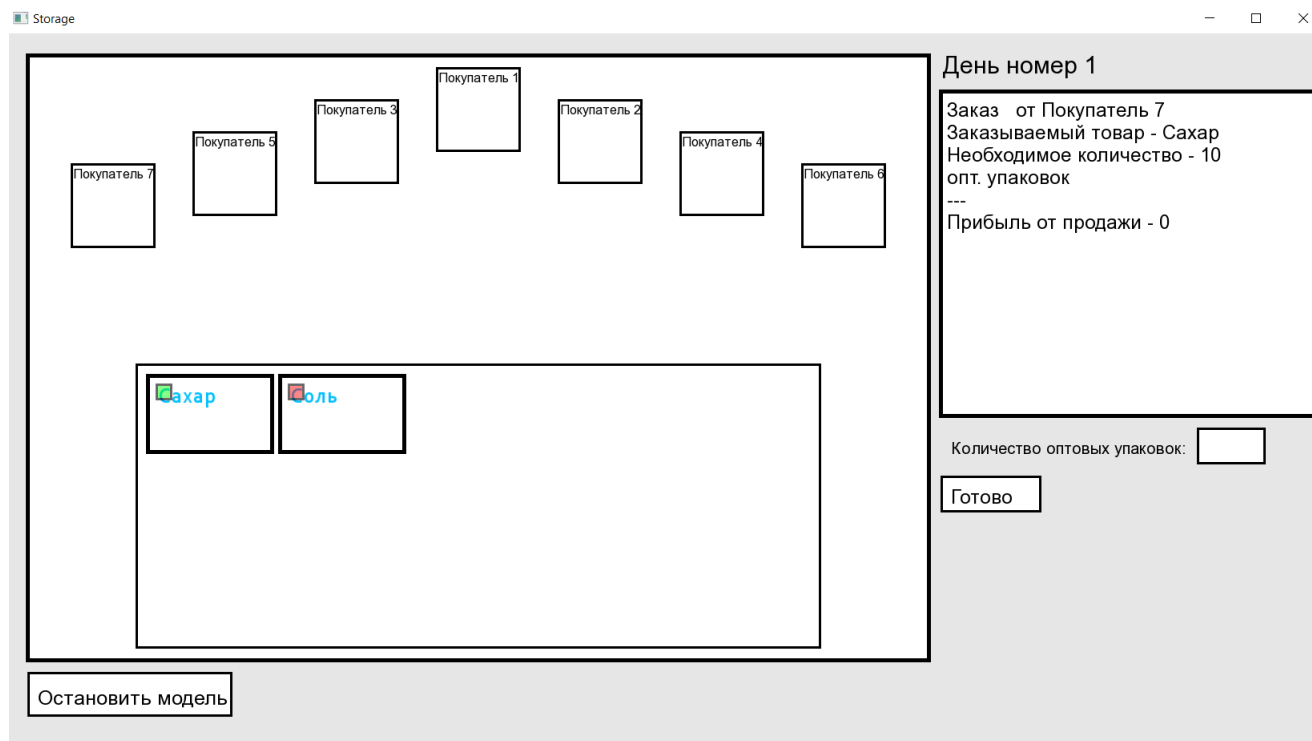
Присутствует анализ ошибок. Например, если пользователь укажет, что есть сахар, но не напишет, сколько его, то модель не запустится.



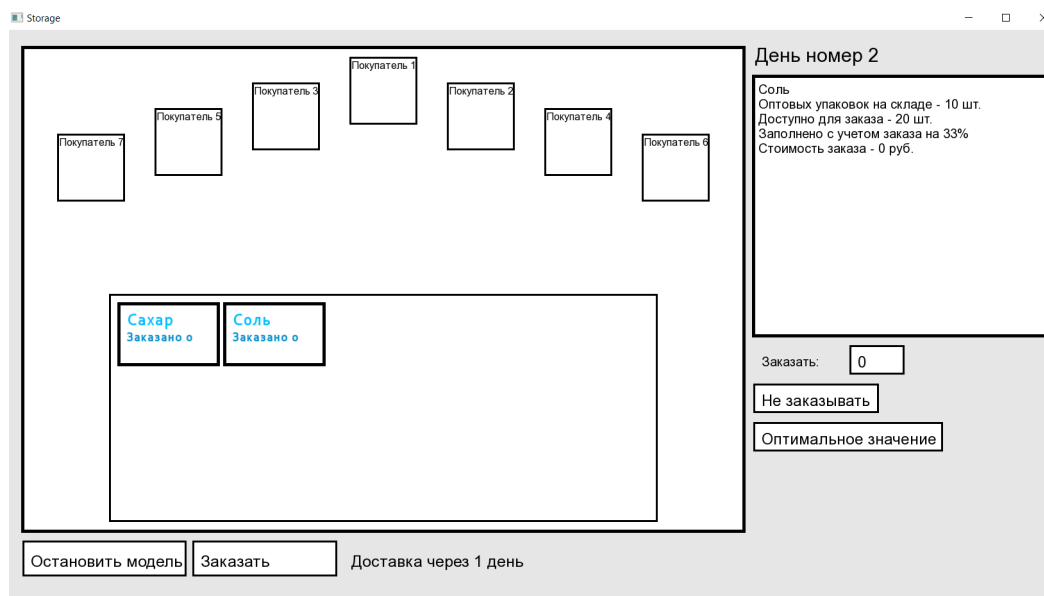
The screenshot shows a window titled "Input model" with standard Windows window controls (minimize, maximize, close). The window contains the following elements:

- Label "Введите кол-во дней:" followed by a text input field.
- Label "Введите кол-во магазинов:" followed by a text input field.
- A list of goods with checkboxes next to them:
 - Сахар
 - Соль
 - Каша
 - Хлопья
 - Макароны
 - Чай
 - Шоколад
 - Кофе
 - Какао
 - Масло
 - Рыба
 - Творог
 - Молоко
 - Сыр
 - Вино
- A button labeled "Запустить" (Run) in the bottom right corner.

Пользователя встречает главное окно модели, в которой есть отображение склада, магазинов, комнат с товарами и партий. Справа от экрана есть информационное поле, в котором пользователь может увидеть конкретную информацию о продуктах на складе, о заказах от магазина, а также ввести кол-во заказываемых товаров.



Чем ближе продукт (точнее, партия продуктов) к своему сроку годности, тем больше его изображение становится красным. Так, пользователю удобно отслеживать испортившиеся товары и уценять их. После пользователь нажимает на кнопку "Принять заказы". Он может посмотреть детали заказа и выбрать, сколько оптовых упаковок он отправит магазину.



После отправки товаров магазину пользователю предоставляется опция заказа товаров на склад. Также снизу будет написано, сколько дней нужно на доставку этих товаров.