

**Колледж космического машиностроения и технологий**

**ОТЧЕТ**

**по экзамену квалификационному СПО**

по профессиональному модулю ПМ.01. «Эксплуатация и модификация информационных систем»

Специальность 09.02.04 «Информационные системы (по отраслям)»

Обучающегося 3 курса группы ИС1-20 формы обучения очной

|  |
| --- |
| **Миловановой Виктории Алексеевны** |
| *(Ф.И.О. полностью)* |

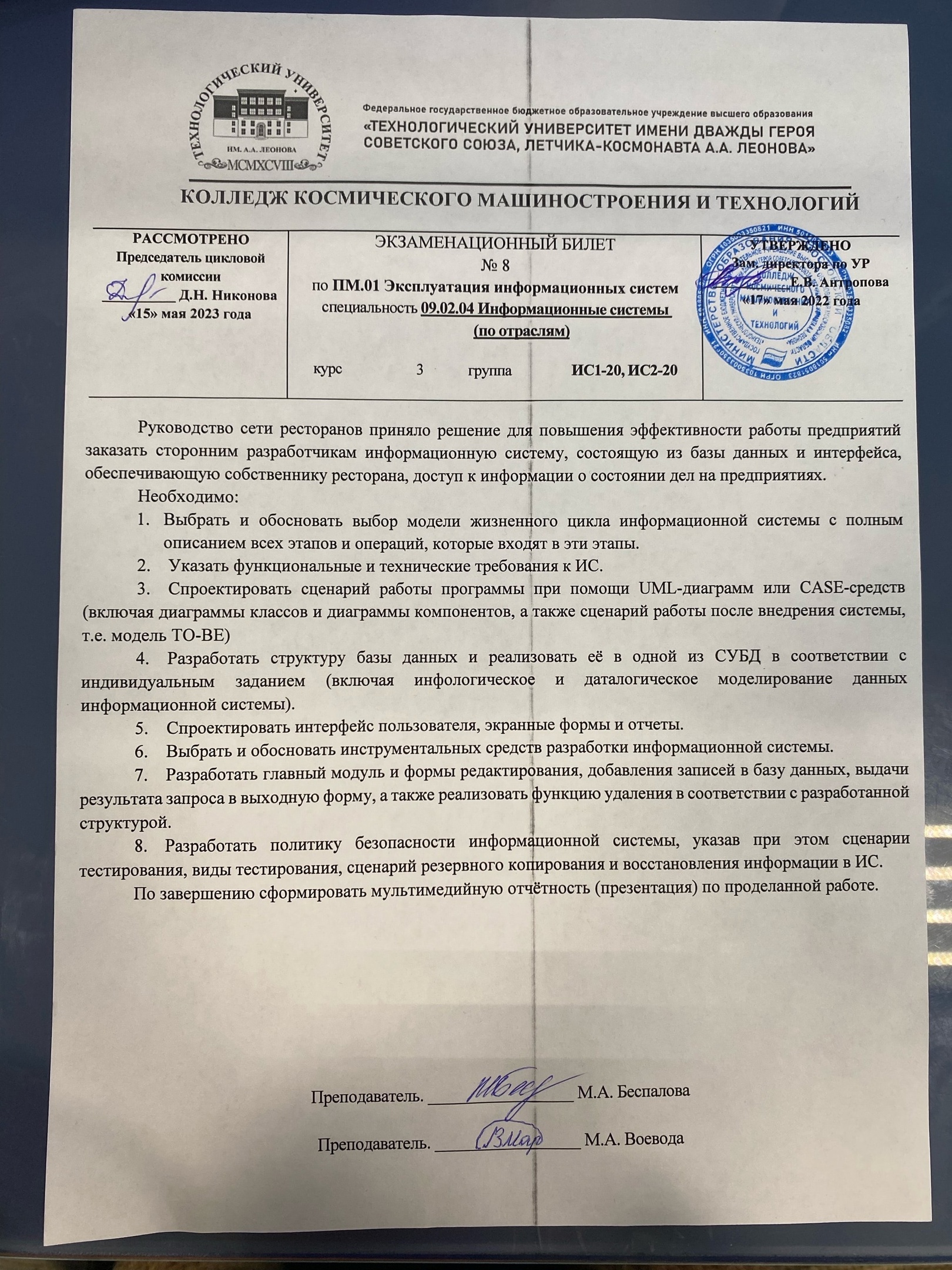
|  |  |  |
| --- | --- | --- |
| Преподаватель |  | М.А. Беспалова |

|  |  |  |
| --- | --- | --- |
| Преподаватель |  | М.А. Воевода |

|  |  |
| --- | --- |
| Итоговая оценка по экзамену |  |

https://github.com/viireneya/Qualification\_exam\_PM\_01

г.Королёв, 2023 г.



Содержание

[1. Жизненный цикл 4](#_Toc139001537)

[2. Функциональные и технические требования 5](#_Toc139001538)

[3. Спроектированный сценарий работы системы 7](#_Toc139001539)

[4. Структура базы данных 9](#_Toc139001540)

[5. Интерфейс 11](#_Toc139001541)

[6. Выбор и обоснование инструментальных средств 13](#_Toc139001542)

[7. Разработка 14](#_Toc139001543)

[8. Политика безопасности 20](#_Toc139001544)

# Жизненный цикл

Для выполнения экзаменационного задания по разработке ИС были изучены модели жизненного цикла, после чего была выбрана спиральная модель, представленная на рисунке 1.

Спиральная модель жизненного цикла - представляет собой процесс разработки программного обеспечения, сочетающий в себе как проектирование, так и повторяющуюся последовательность циклов разработки с непрерывным контролем рисков.

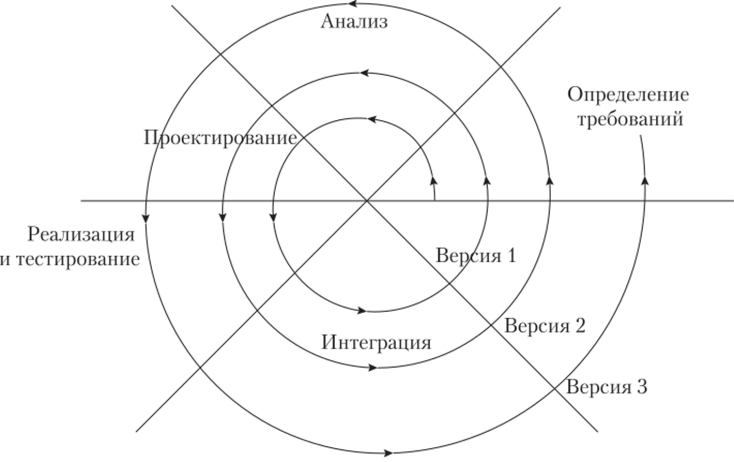


Рисунок 1. Спиральная модель жизненного цикла

Спиральная модель была выбрана по следующим причинам: она позволяет добавлять дополнительный функционал к программному обеспечению на самых поздних стадиях разработки, постоянно контролировать процесс, проводя регулярные экспертизы текущего состояния проекта, что является неотъемлемой частью данной модели жизненного цикла, общее видение проекта становится более ясным, существует возможность вернуться к предыдущим фазам в случае необходимости пересмотра принятых решений.

# Функциональные и технические требования

Определены следующие требования:

* Добавлять новые записи в документах, справочниках;
* Позволять редактировать записи в документах и справочниках;
* Позволять удалять записи и данные в документах и справочниках, за исключением случая, если какой-то элемент должен быть обязательно заполнен;
* Отслеживать движение продукции, начиная с закупки продукции и заканчивая продажей;
* Отслеживать количество продаваемых блюд по организациям, используя отчет в виде диаграммы;
* Отслеживать количество имеющейся продукции на складах, используя отчет остатков продукции;
* Отслеживать состояние организаций, используя справочники;
* Создавать и редактировать блюда, которые подаются в организациях.

Технические требования:

* Процессор Intel Pentium Celeron 2700 Мгц и выше;
* Оперативная память 4 Гбайт и выше;
* Жесткий диск 40Гб и выше;
* Видеокарта, поддерживающая монитор с минимальным разрешением 1280x768 точек.

Системные требования отображены на рисунках 2-4:

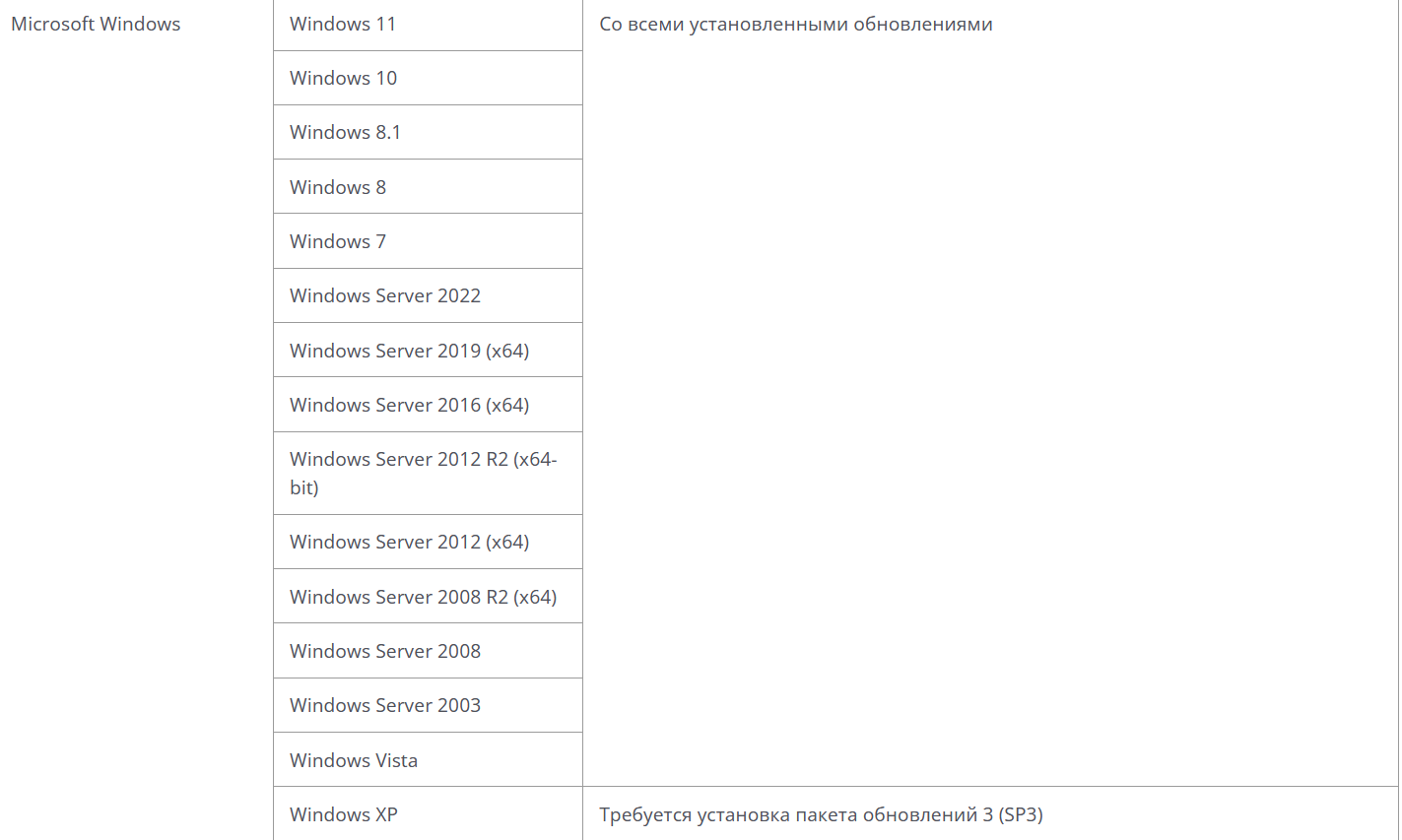


Рисунок 2. Системные требования (Windows)

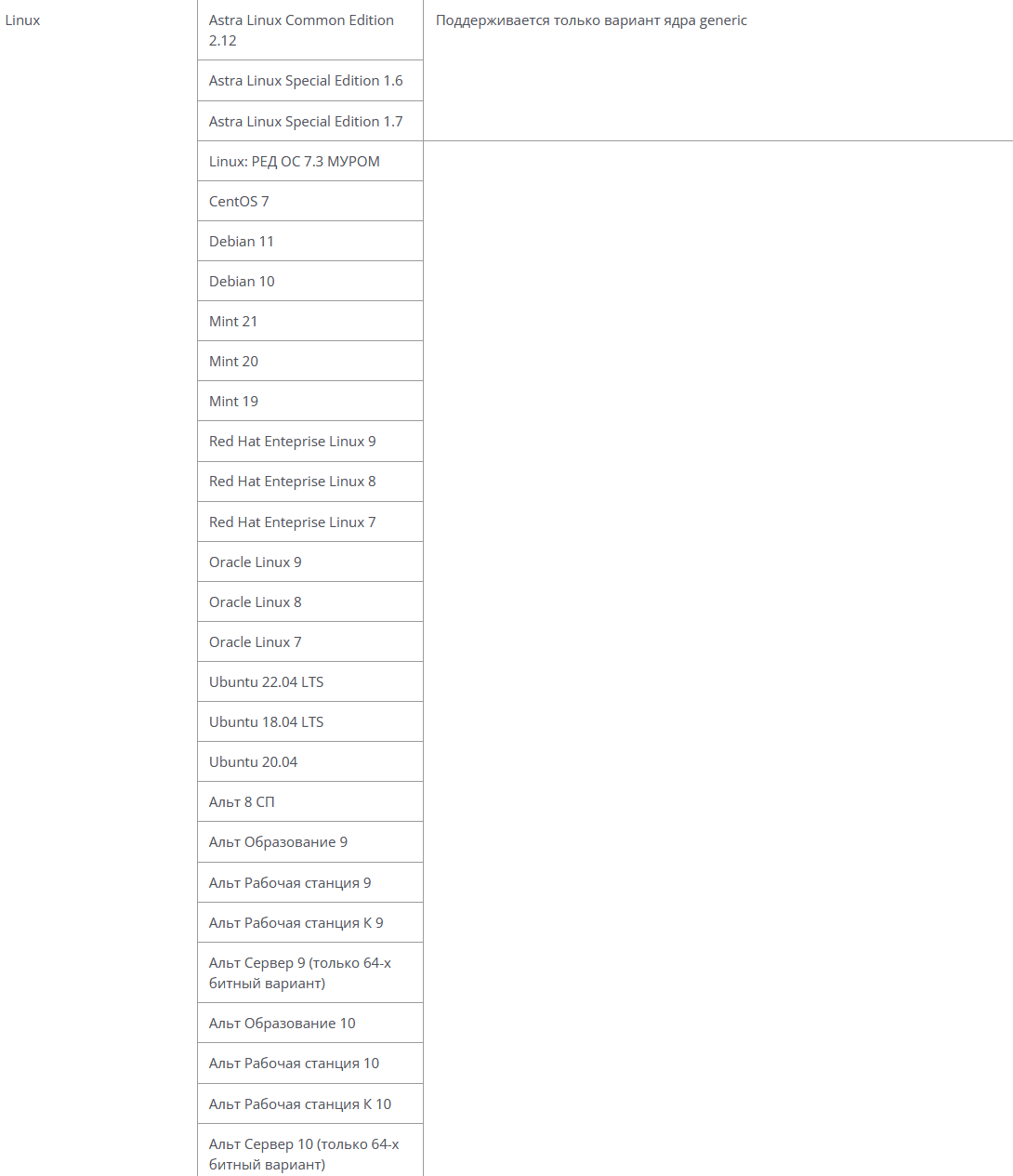


Рисунок 3. Системные требования (Linux)



Рисунок 4. Системные требования (macOS)

# Спроектированный сценарий работы системы

TO-BE

Моделирование TO-BE – это процесс разработки бизнес-процесса, которое должно быть оптимальным и эффективным, определяются изменения, которые необходимо внести в текущий процесс, чтобы достичь более эффективного состояния. Модель TO-BE изображена на рисунке 5.

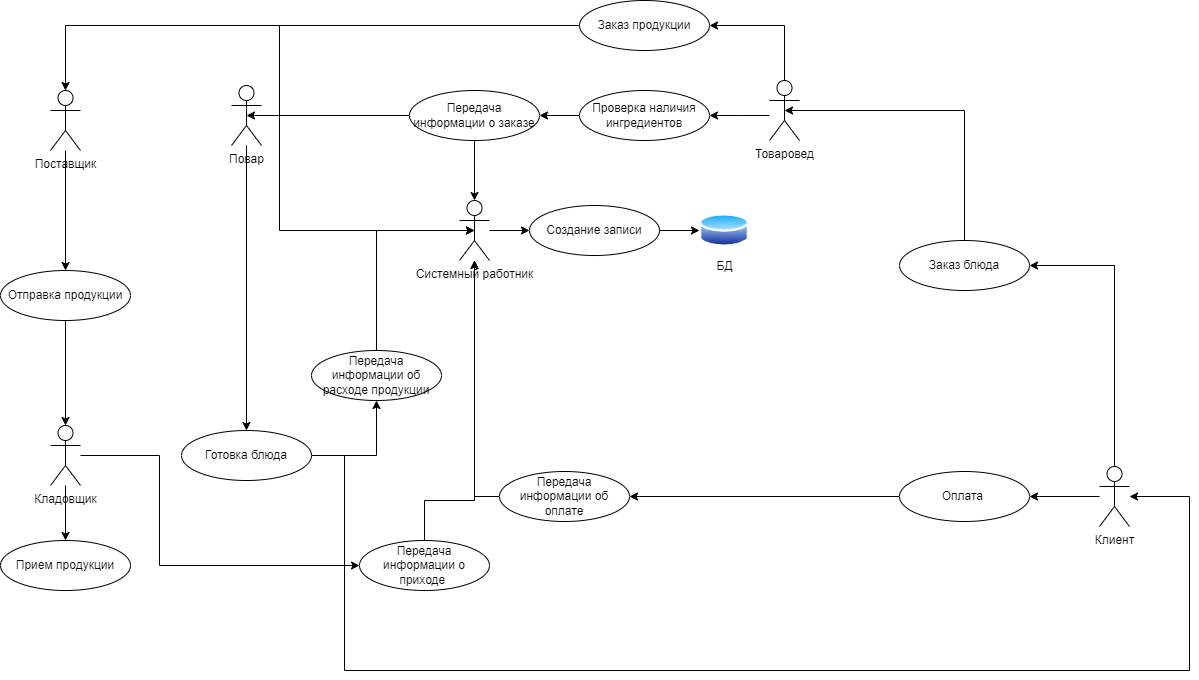


Рисунок 5. Модель TO-BE бизнес-процесса деятельности ресторанов

В бизнес-процессе учувствуют следующие акторы:

* Поставщик;
* Клиент;
* Кладовщик;
* Товаровед;
* Повар;
* Системный работник.

Клиент приходит в ресторан, выбирает блюдо, совершает заказ. Данные передаются товароведу, в свою очередь он проверяет наличие продукции на складе в данной организации. После проверки товаровед передает информацию системному работнику, который создает запись в БД, и повару с его командой. Повар приготавливает блюда, после чего системному работнику передается информация о расходе ингредиентов, он создает запись в БД. После получения блюда, клиент оплачивает заказ, данные также передаются системному работнику, который создает запись в БД о получении денег. Товаровед регулярно проверяет наличие ингредиентов, если ингредиентов не хватает, он заказывает продукцию у поставщика, данные одновременно переходят системному работнику, он создает запись в БД. После того, как поставщик отправил продукцию, кладовщик принимает продукцию и передает информацию системному работнику, который записывает данные в БД.

# Структура базы данных

Для разработки ИС необходимо составить инфологическую модель. Данная модель позволяет логически описать структуру, представления данных и их отношений. Инфологическая модель представлена на рисунке 6.

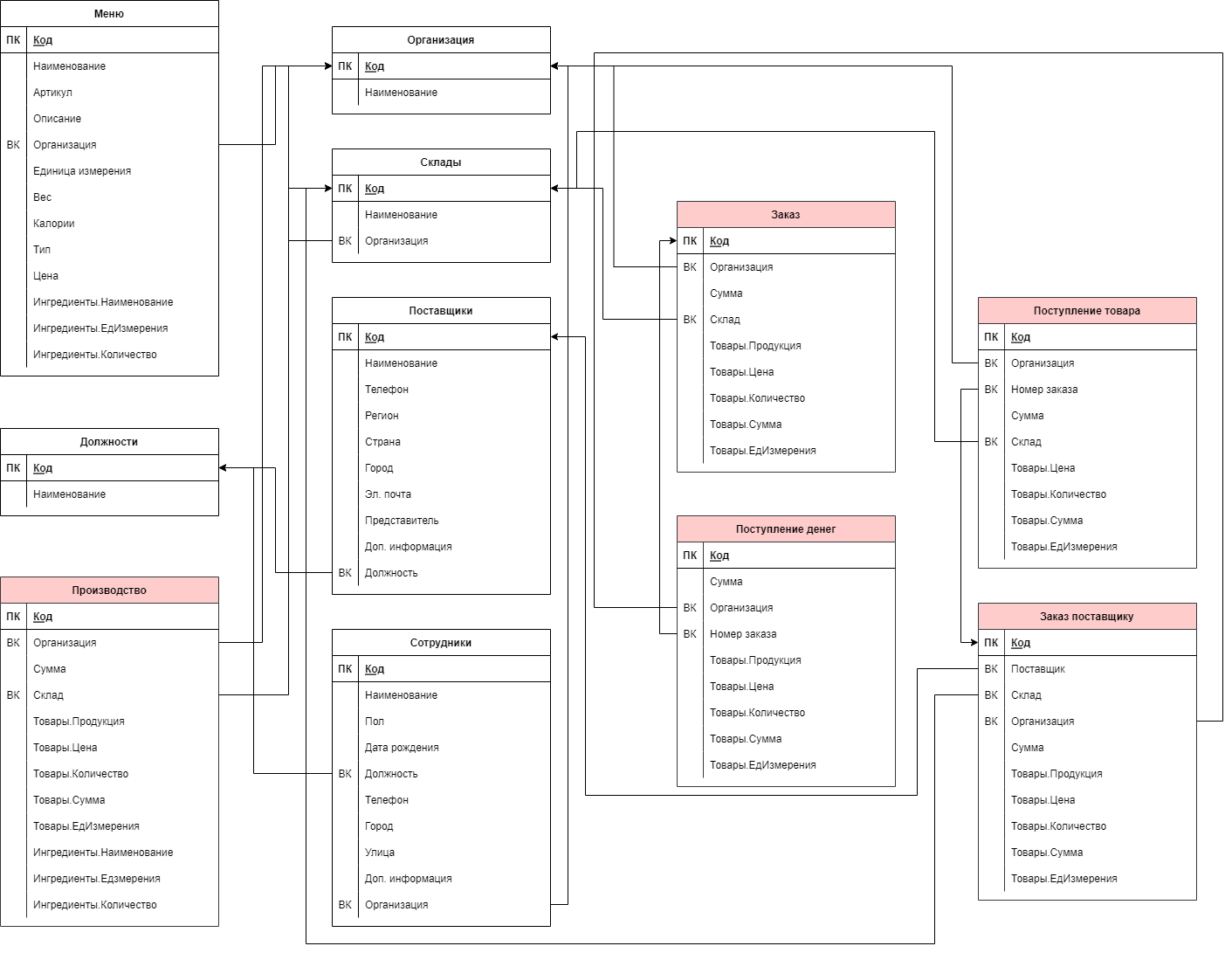


Рисунок 6. Инфологическая модель

Даталогическая модель представляет собой одним из основных пунктов при проектировании базы данных. Она помогает понять разработчикам структуру, тип данных системы. Данная модель служит основой для создания физической модели. Связи такие же, как и в инфологической модели. Даталогическая модель представлена на рисунок 7.

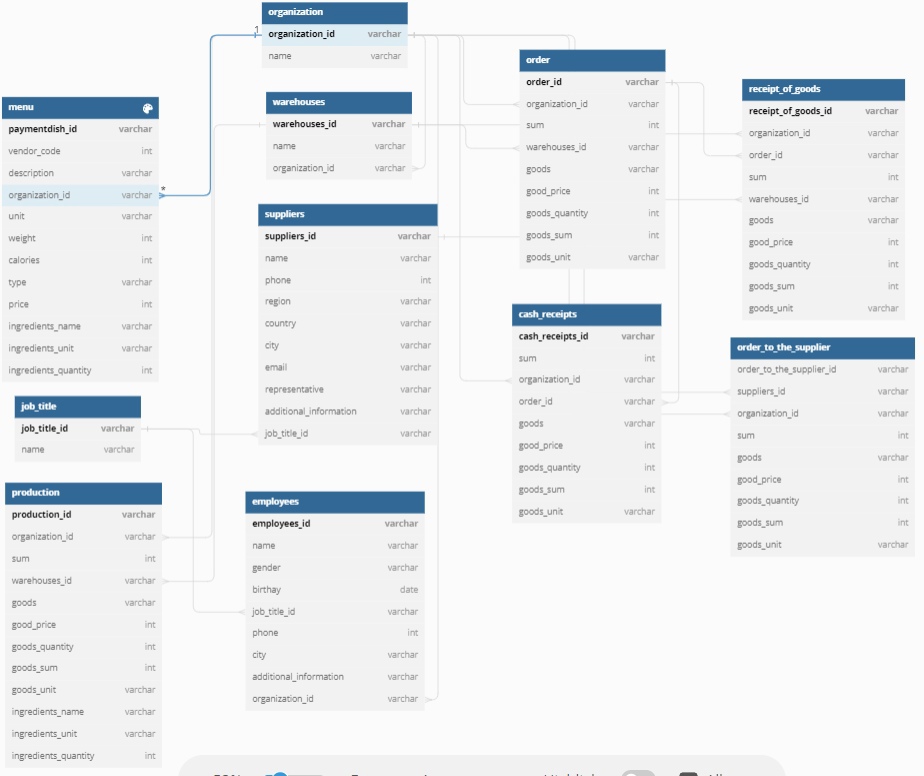


Рисунок 7. Даталогическая модель

# Интерфейс

Для проектирования графического интерфейса создаются макет. Для этого важно понимать потребности и ожидания пользователей системы. Макеты должны быть интуитивно понятными, а для этого нужно изучить интерфейсные элементы, необходимые для работы. На этом этапе проектируется общая структура интерфейса, включая расположение элементов управления, навигацию, оформление и цветовую схему. Макеты могут быть созданы с использованием специальных инструментов, таких как графические редакторы или специализированные инструменты проектирования интерфейса.

Справочники служат основным объектом системы, все они сделаны по одному макету и не меняются. Макет справочников представлен в таблице 1.

Таблица 1. Макет справочника

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование | | | | |
| **Записать и закрыть** | Записать | |  | |
| Код: | |  | |  |
| Тип: | |  | | |
| Наименование: | |  | | |
| Артикул: | |  | |  |
| Описание: | |  | | |
| Единица измерения: | |  | | |
| Вес, количество: | |  | |  |
| Калории: | |  | |
| Цена: | |  | |

Документ является усовершенствованным справочником, внутри находятся строки с общими данные и табличные части. Интерфейс у документов не отличаются, все они придерживаются одному макету, представленном в таблице 2.

Таблица 2. Макет документа

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Наименование | | | | | | | | | | |
| **Провести и закрыть** | | Записать | | Провести | | Создать на основании | | | |  |
|  | | | | | | | | Сумма |  | |
| Номер |  | | | | Дата | |  | |  | |
| Организация |  | | | | | | |  | | |
| Добавить |  | | | | | | | | | |
| N | Блюдо | | Количество | | | | Цена за ед. | | Сумма | |
|  |  | |  | |  | |  |  |  | |

Регистр накопления служит для выдачи дополнительной информации о каком-то реквизите. Графический интерфейс у регистров накопления не меняется, макет регистра представлен в таблице 3.

Таблица 3. Макет регистра накопления

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название | | | | | |
| Период | Регистратор | Номер строки | Продукция | Организация | Количество |
|  |  |  |  |  |  |

Отчеты служат объектом для вывода нужной информации для передачи в бухгалтерию или другие отделы. Отчет берет информацию из справочника или документа. Интерфейс у отчетов табличного вида одинаковый, макет представлен в таблице 4.

Таблица 4. Макет отчета

|  |  |  |  |
| --- | --- | --- | --- |
| **Продукция** | **Организация1** | **Организация2** | **Итого** |
| **Количество остаток** | **Количество остаток** | **Количество остаток** |
| Продукция1 |  |  |  |
| **Итого** | **Сумма** | **Сумма** | **Сумма** |

# Выбор и обоснование инструментальных средств

Для выполнения экзаменационного задания было принято решение разрабатывать информационную систему на платформе 1С: Предприятие.

Система программ 1С: Предприятие состоит из технологической платформы (ядра) и разработанных на ее основе прикладных решений (конфигураций). Такая архитектура системы принесла ей высокую популярность, поскольку обеспечивает открытость прикладных решений, их функциональность и гибкость, короткие сроки внедрения, высокую производительность, масштабируемость от одного до десятков тысяч рабочих мест, работу в режиме «облачного» сервиса и на мобильных устройствах.

Программы 1С просты в освоении. Это относиться, как для конечных пользователей, так и для разработчиков. Не смотря на множество различных приложений на платформе 1С они все сделаны по одним правилам. Поэтому если освоить одну программу, то другие будет изучать гораздо быстрее и легче.

Современные решения на платформе 1С позволяют одновременно обслуживать тысячи пользователей, которые работают с различных устройств под разными операционными системами, обеспечивать бесперебойную работу больших производственных предприятий и все это не могло быть без должного уровня качества платформы 1С.

# Разработка

В 1С: Предприятии справочники и документы представляют собой таблицы с данными. В конфигураторе при создании справочника, документа или иного объекта можно настроить созданный объект через специальное меню. Перемещаясь по разделам, можно углубленно настроить каждый аспект объекта.

В 1С: Предприятие справочники представляют собой таблицы с данными, его можно настроить через специальное меню. Пример справочника изображен на рисунке 8.

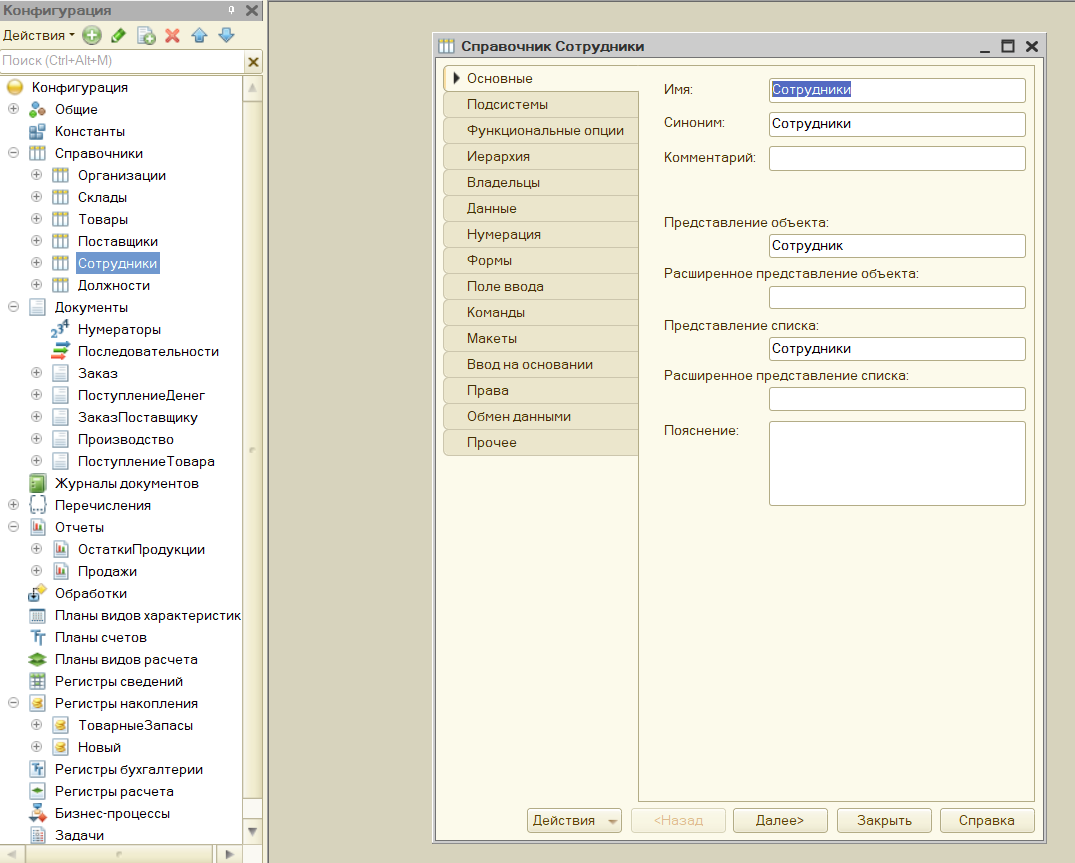


Рисунок 8. Пример справочника "Сотрудники"

Справочник можно настраивать, улучшать и усложнять, перемещаясь по меню справочника. В подсистеме необходимо выбрать в какой подсистеме будет отображаться справочник. Данные – важная часть справочника, так как именно туда добавляются данные, с которыми будет взаимодействовать в системе пользователь. Формы – вторая важная часть справочника, здесь настраивается то, как в системе будет выглядеть та или иная форма справочника. Пример на рисунке 9.

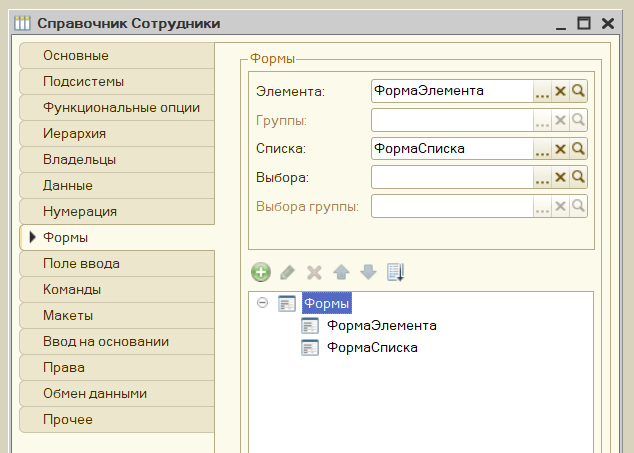


Рисунок 9. Формы в справочниках

Права – здесь настраивается права доступа и взаимодействия для отдельных пользователей системы.

На рисунке 10 представлен справочник «Сотрудники» в системе.

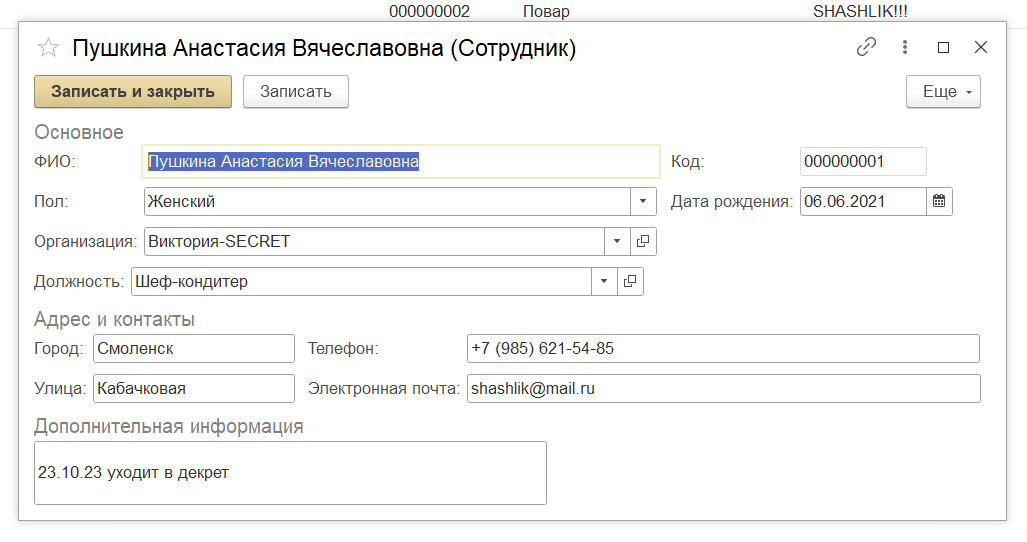


Рисунок 10. Справочник "Сотрудники" в системе

Документы в 1С: Предприятие. В документах также необходимо выбрать подсистему, и, конечно же, заполнить данные, необходимые для документа. Пример приведен на рисунке 11.

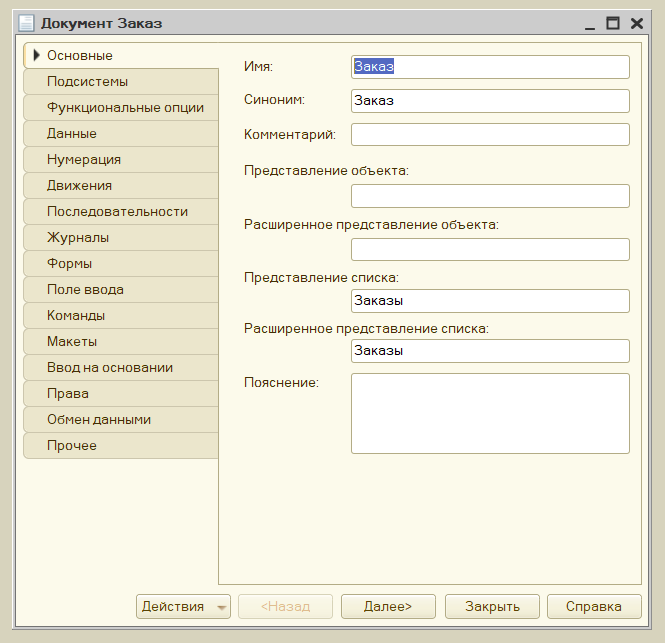


Рисунок 11. Пример документа

Также в данном документе заполняются формы. В самой работе с системой данный документ «Заказ» представлен на рисунке 12.

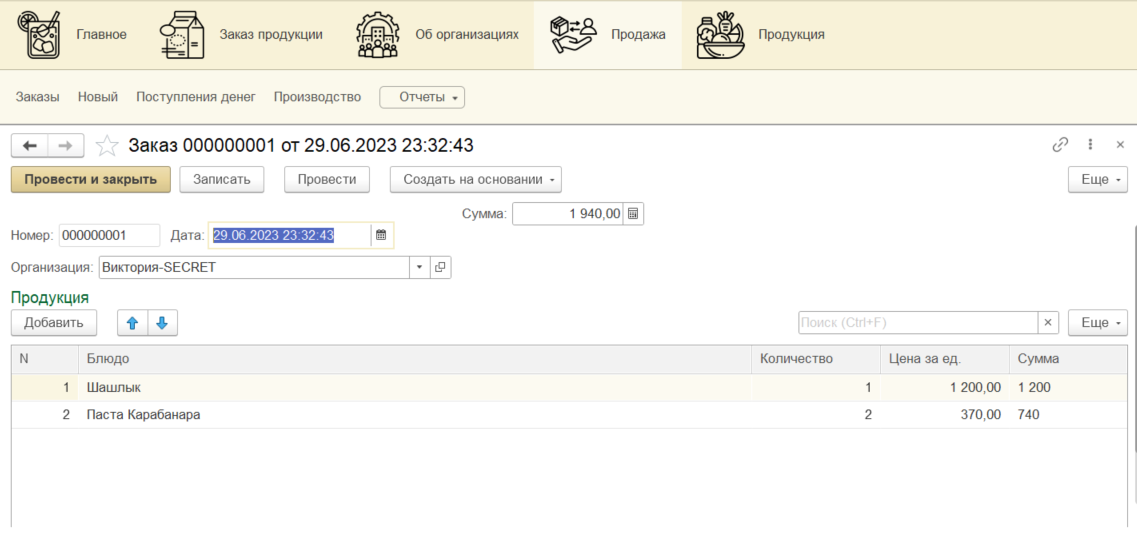


Рисунок 12. Документ "Заказ"

Пример схемы компоновки данных в отчете системеы на примере отчета «Остатки продукций», он представлен на рисунке 13.

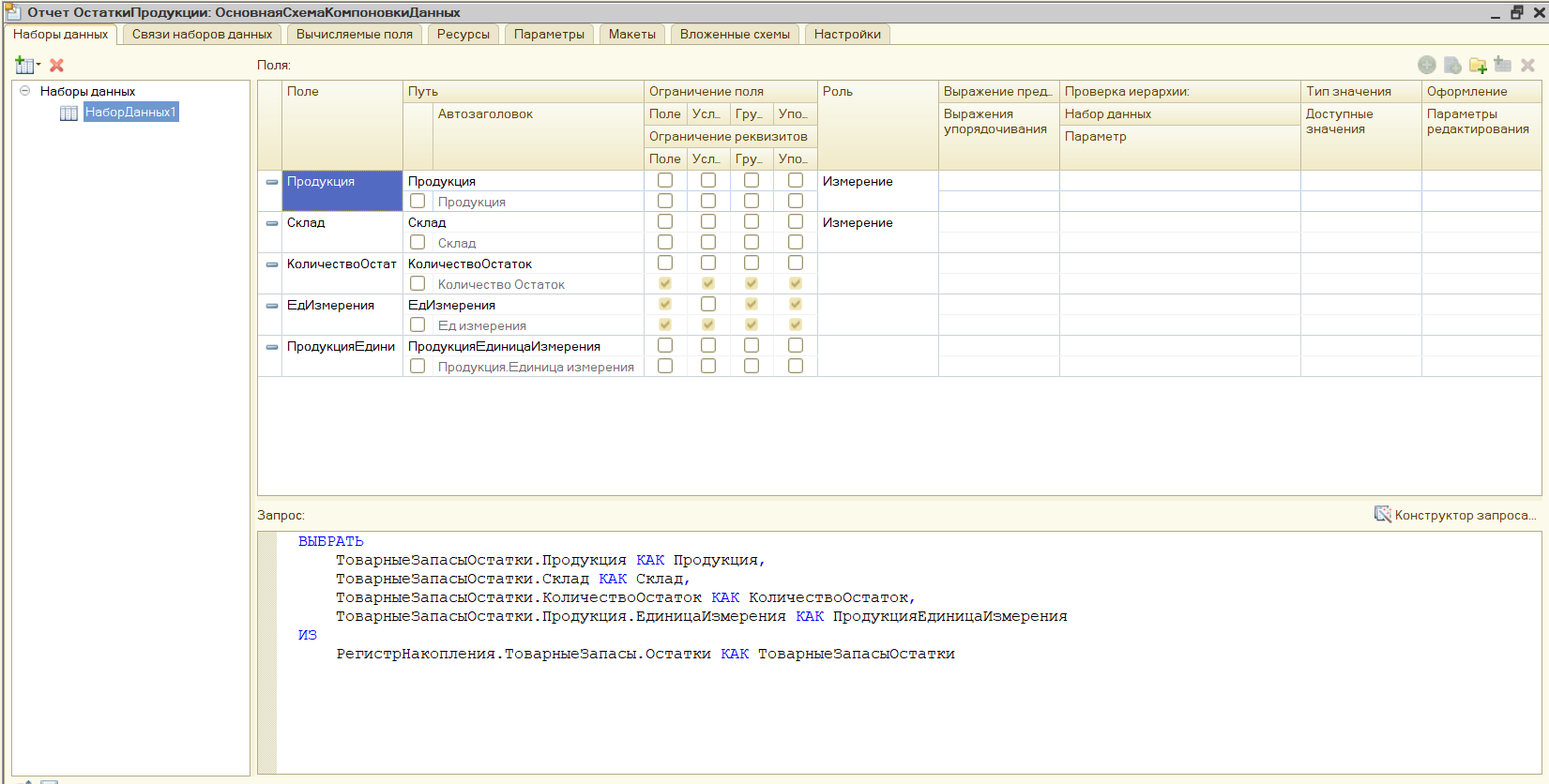


Рисунок 13. Схема компоновки данных отчета "Остатки продукций"

На рисунке 14 представлено, как отчет выглядит в системе.

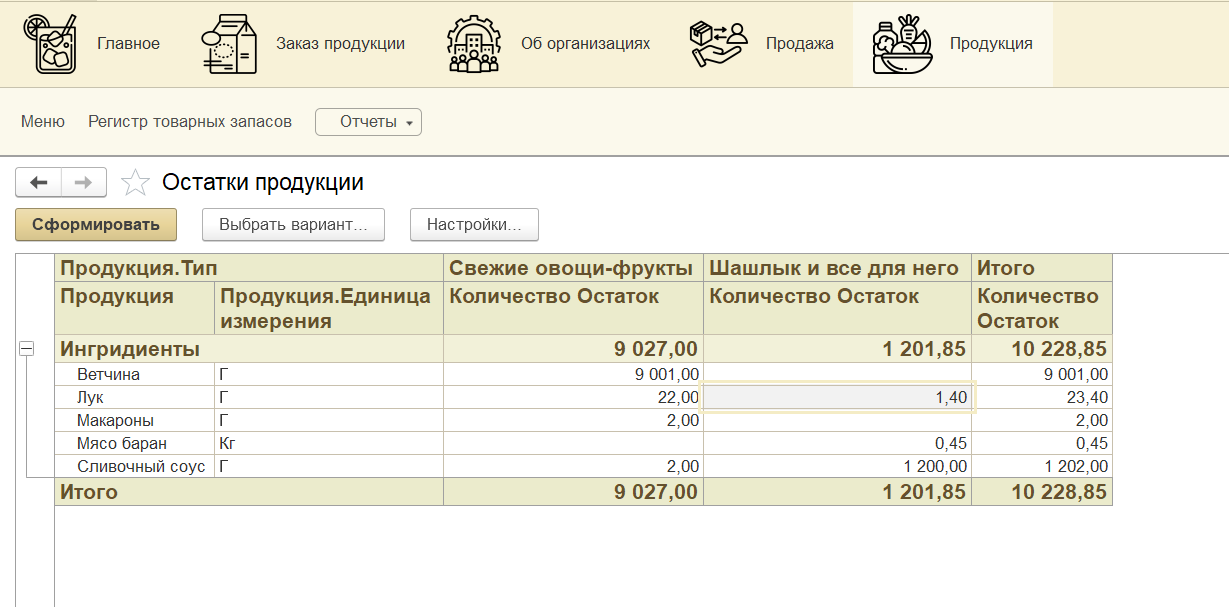


Рисунок 14. Отчет "Остатки продукции"

# Политика безопасности

Резервное копирование и восстановление:

В 1С есть несколько способов резервного копирования. Рассмотрим самый распространенный:

Открывается конфигуратор, затем необходимо нажать: «Администрирование» - «Выгрузить информационную базу». После чего выбираете место для сохранения и получаете dt-файл.

Этот файл представляет из себя не только данные базы, но и саму конфигурацию, что очень удобно, так как это полезно в процессе разработки, а также, такое резервирование поможет в случае случайных изменений в конфигурации неопытным пользователем.

Восстановление, основанное на методе резервного копирования, рассмотренном ранее:

Открывается конфигуратор, затем необходимо нажать: «Администрирование» - «Загрузить информационную базу». Открывается окно выбора dt-файла – указываем заранее резервированную базу и нажимаете «Открыть». После нажимаете «Подтвердить», конфигуратор перезапускается – база восстановлена.

Тест кейс «Добавление сотрудника»

|  |  |
| --- | --- |
| **Действие** | **Ожидаемый результат** |
| Нажать в меню на подсистему «Об организациях» | Открылась форма с справочниками об организациях |
| Нажать на справочник «Сотрудники» | Открылся справочник «Сотрудники», с списком сотрудников |
| Нажать на кнопку «Создать» | Открылась форма заполнения с данными, нужными для сотрудника |
| Корректно заполнить все данные | Ошибок не возникает |
| Нажать кнопку «Записать и закрыть» | Форма закрылась, новый сотрудник появился в списке |

Тест кейс «Оформление поступления товаров»

|  |  |
| --- | --- |
| **Действие** | **Ожидаемый результат** |
| Нажать в меню на подсистему «Заказ продукции» | Открылась форма с справочниками и документами |
| Нажать на справочник «Заказ продукции» | Открылся документ «Заказ продукции», с списком заказов у поставщика |
| Выделить нужный заказ и нажать на кнопку «Поступление товара: на основании» | Открылся документ поступления товаров, куда занеслись все данные |
| Нажать на кнопку «Провести и закрыть» | Ошибок не возникает, форма закрывается, поступление товара произошло |

Тест кейс «Добавление ингредиента»

|  |  |
| --- | --- |
| **Действие** | **Ожидаемый результат** |
| Нажать в меню на подсистему «Продукция» | Открылся справочник «Меню» |
| Нажать два раза для открытия на «Ингредиенты» | Открылся список уже существующих ингредиентов |
| Нажать на кнопку «Создать» | Открылась форма создания ингредиента |
| Корректно заполнить данные | Ошибок не возникает |
| Нажать «Записать и закрыть» | Новый ингредиент создался |