**Введение**

Программа «1С» — это продукт для автоматизации предприятий. Перед пользователями открыт широкий ассортимент типовых решений и отраслевых конфигураций. С их помощью возможно автоматизировать бизнес-процессы в любой компании, независимо от масштабов и направлений. Почти за три десятилетия ИТ-решения выросли из простой программы для ведения учета и сдачи отчётности до многопользовательских систем класса ERP.

Все программные решения фирмы «1С» созданы на базе единой технологической платформы «1С:Предприятие».

Области, где для автоматизации применяются программы «1С:Предприятие»:

* Бухгалтерский и налоговый учет по стандартам России и стран СНГ, отчетность МСФО;
* Торгово-складской учет и управление торговой компанией;
* Управленческий учет и бюджетирование, консолидация;
* Учет денежных средств и финансовое планирование;
* Управление производством и учет производственных затрат, расчет себестоимости;
* Управление логистикой, закупками;
* Розничная торговля;
* [Управление торговлей на маркетплейсах.](https://www.1cbit.ru/1csoft/bit-integracia-marketplace/?utm_source=statiya_chto-takoe-programma-1s" \t "https://www.1cbit.ru/blog/chto-takoe-programma-1s/_blank)
* Управление документооборотом;
* Кадровый учет и управление персоналом, ведение кадровой политики и расчетов с сотрудниками;
* Управление взаимоотношениями с клиентами (CRM);
* Комплексное управление предприятиями (ERP) различных сфер деятельности, в т.ч. – производство, торговля, строительство, сельское хозяйство, ТЭК, ЖКХ, финансовый сектор, общественное питание, гостиничный бизнес, медицина и пр.

**1. Постановка задачи**

С помощью 1C нужно спроектировать программу для учета дел бизнеса по продаже мебели. Информация о расходах, доходах и продажах.

**2. Проектирование интерфейса приложения**

В программе 1С была создана БД об учете по продаже мебели, для этого в самом 1C была создана рабочая область для учета продажи, остатков и прихода мебели. В конфигураторе были созданы справочники, константы, документы, журналы документов и другие объекты.

* 1. Правила и принципы разработки интерфейса

1. Для каждой укрупненной группы пользователей (рабочего места) в конфигурации рекомендуется определять отдельный интерфейс: главное меню, набор и состав панелей инструментов.

2.1. Главное меню обеспечивает доступ ко всем формам, которые требуются пользователю для работы или сервисных функций. В подменю с большим числом элементов, группы элементов должны быть ограничены разделителями. Максимальное число элементов в одной группе не более 7.

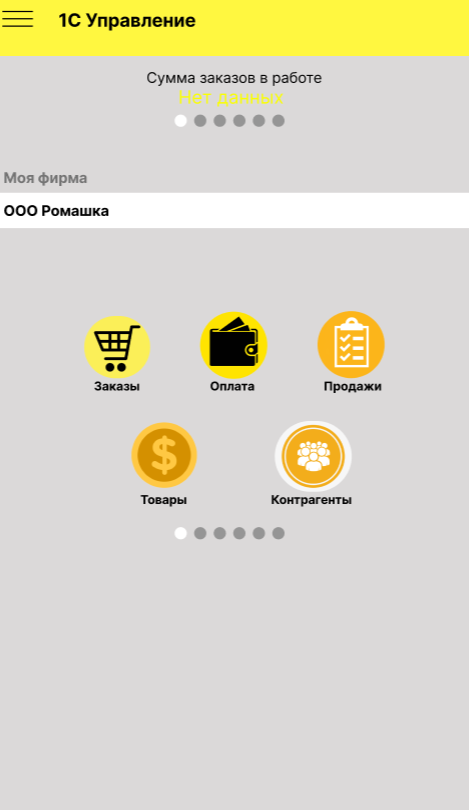
2.2. При любом положении выбора действия из главного меню, его размер должен быть таким, чтобы полностью умещаться на экране при минимальном разрешении, из расчета на которое разработана конфигурация.

2.3. По умолчанию пункты меню для доступа к справочникам, документам, планам видов характеристик, планам счетов, планам видов расчетов и регистрам вызывают формы списков этих объектов.

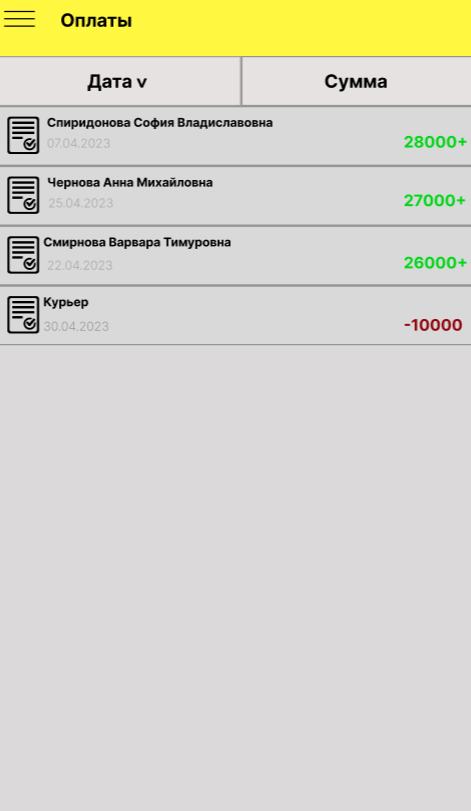
3.1. Интерфейс следует проектировать таким образом, чтобы группе пользователей, с одной стороны, был доступен необходимый набор действий, а с другой, не предоставлялся доступ к действиям на которые нет прав. Вызовы наиболее часто выполняемых пользователем действий в интерфейсе лучше располагать так, чтобы они были наиболее доступны, и наоборот.

3.2. Желательно однотипные блоки меню и панелей инструментов в разных интерфейсах делать похожим образом.

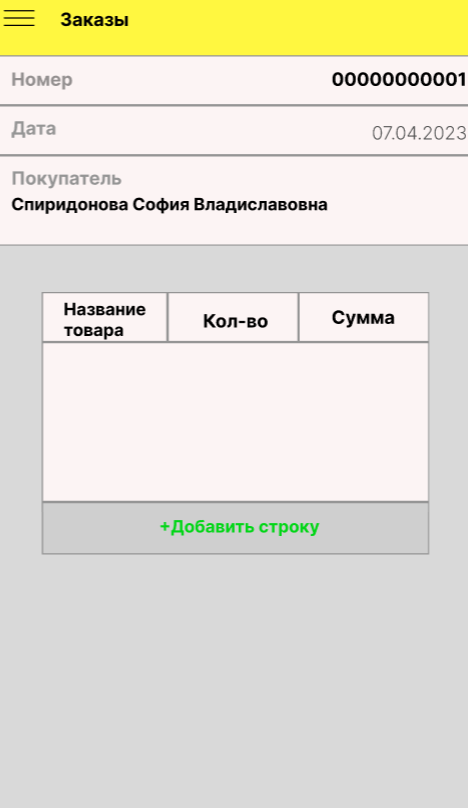
2.2. Разработка макета приложения



(Рис.1 “Главная страница”)



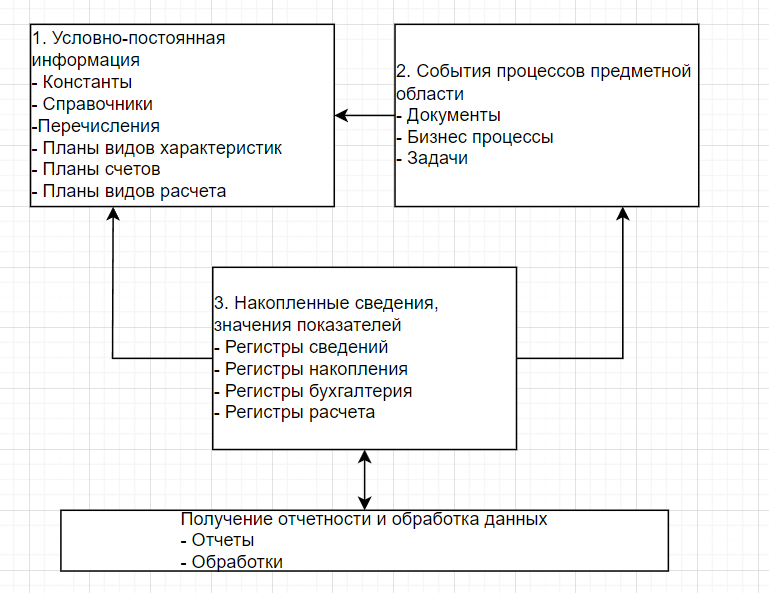
(Рис.2 “Оплата”)



(Рис.3 “Заказы”)

1. Структура хранения данных

* При проектировании системы одной из задач является выбор типов объектов метаданых для реализации хранения соответствующих сущностей предметной области. Неправильный выбор типов объектов ведет к неэффективности прикладного решения, невозможности его последующего развития и делает невозможным адаптацию к возможным изменениям состава решаемых задач.
* При выборе типа объектов метаданных в общем случае следует руководствоваться следующей схемой прикладного решения



(Рис.4 ”Схема структур хранения данных”)

На схеме выделяются следующие блоки:

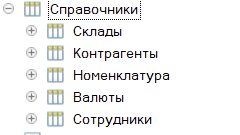
1. ***Условно-постоянная информация***. К этой части относится информация, которая вводится один раз, сравнительно редко изменяется и многократно используется. Примером такой информации могут служить различные классификаторы, настройки, перечни, реестры, нормативно-справочная информация и т.п.

2. *Различные****события процессов предметной области****,* которые привязаны ко времени и могут порождать при регистрации различные сведения, изменять значения показателей. Пример – документооборот предприятия, ведение учета; регистрация заявок, звонков и т. п.

3. ***Накопленные сведения, значения показателей****,* которые характеризуют процессы и текущее состояние прикладной области. В отличие от первых двух частей, эти данные имеют необъектную природу и не являются самостоятельными сущностями с точки зрения прикладной области. Пример – история продаж товара, остатки на складах, текущий бухгалтерский баланс, история изменения курсов валют и т.п.

3.1. Проектирование структуры хранения данных

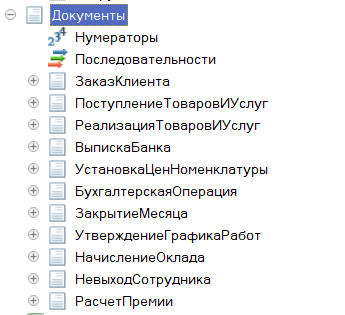
Разрабатывая базу данных на клиент-серверной платформе возникает необходимость быстро и оперативно готовить копию базы без лишней информации (для внешних программистов например). Разрабатывая базу данных на клиент-серверной платформе возникает необходимость быстро и оперативно готовить копию базы без лишней информации (для внешних программистов например). Данная обработка позволяет по названиям объектов метаданных быстро очистить таблицы или колонки в базе данных SQL — прямым запросом.



(Рис.5 “Справочники”)

3.2. Наполнение структуры данными

Структура в 1С это динамический набор данных (коллекция значений), каждый элемент которой состоит из пары «Ключ» и «Значение». Ключи структуры уникальны, поэтому ими можно идентифицировать значения. Ключ структуры должен иметь строковый тип данных и отвечать требованиям к именам переменных. К значениям структуры можно обращаться как к свойствам объекта, при этом ключ используется как имя свойства

.

(Рис.6 “Документы”)

Пример кода **Заказ клиента:**

Процедура ПечатьСчетаНаОплату(ТабДок, Ссылка) Экспорт

//{{\_КОНСТРУКТОР\_ПЕЧАТИ(ПечатьСчетаНаОплату)

Макет = Документы.ЗаказКлиента.ПолучитьМакет("ПечатьСчетаНаОплату");

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ЗаказКлиента.Дата,

| ЗаказКлиента.Клиент,

| ЗаказКлиента.Номер,

| ЗаказКлиента.Склад,

| ЗаказКлиента.Товары.(

| НомерСтроки,

| Товар,

| Количество,

| Цена,

| Сумма,

| Склад

| )

|ИЗ

| Документ.ЗаказКлиента КАК ЗаказКлиента

|ГДЕ

| ЗаказКлиента.Ссылка В (&Ссылка)";

Запрос.Параметры.Вставить("Ссылка", Ссылка);

Выборка = Запрос.Выполнить().Выбрать();

ОбластьЗаголовок = Макет.ПолучитьОбласть("Заголовок");

Шапка = Макет.ПолучитьОбласть("Шапка");

ОбластьТоварыШапка = Макет.ПолучитьОбласть("ТоварыШапка");

ОбластьТовары = Макет.ПолучитьОбласть("Товары");

Подвал = Макет.ПолучитьОбласть("Подвал");

ТабДок.Очистить();

ВставлятьРазделительСтраниц = Ложь;

Пока Выборка.Следующий() Цикл

Если ВставлятьРазделительСтраниц Тогда

ТабДок.ВывестиГоризонтальныйРазделительСтраниц();

КонецЕсли;

ТабДок.Вывести(ОбластьЗаголовок);

Шапка.Параметры.Заполнить(Выборка);

ТабДок.Вывести(Шапка, Выборка.Уровень());

ТабДок.Вывести(ОбластьТоварыШапка);

ВыборкаТовары = Выборка.Товары.Выбрать();

Пока ВыборкаТовары.Следующий() Цикл

ОбластьТовары.Параметры.Заполнить(ВыборкаТовары);

ТабДок.Вывести(ОбластьТовары, ВыборкаТовары.Уровень());

КонецЦикла;

Подвал.Параметры.Заполнить(Выборка);

ТабДок.Вывести(Подвал);

ВставлятьРазделительСтраниц = Истина;

КонецЦикла;

//}}

КонецПроцедуры

4. Разработка технической документации