Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

Кафедра информационных систем и программной инженерии

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к выпускной квалификационной работе

«Информационная система барбершоп»

Выполнил:

Студент группы ИСсп-117

Раков А. Н.

Принял:

?

Владимир, 2021

**АННОТАЦИЯ**

Тема: Разработка информационной системы «Барбершоп».

Ключевые слова: информационная система, барбершоп, расписание, услуга.

В данной работе представлена реализация взаимодействия расписания мастеров барбершопа с конечным клиентом, написанная с использованием WEB технологий.

Выпускная квалификационная работа представлена на 66 страницах, рисунков – 15, использованных источников – 8, приложений – 3, таблиц - 3.

**ABSTRACT**

Subject: Development of information system «Barbershop».

Keywords: information system, barbershop, schedule, service.

This paper presents the implementation of the interaction of the barbershop master schedule with the end client, written using WEB technologies.

The course project is presented on 66 pages, figures-15, used sources-8, applications-3, tables-3.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 2](#_Toc27480356)

[1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 3](#_Toc27480357)

[1.1 Описание предметной области 3](#_Toc27480358)

[1.2 Цели и задачи разработки 3](#_Toc27480359)

[1.3 Процессы предприятия 4](#_Toc27480360)

[1.4 Описание мобильной конфигурации 6](#_Toc27480361)

[2 РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ 9](#_Toc27480362)

[2.1 Составляющие информационной системы 9](#_Toc27480363)

[2.2. Описание взаимодействия конфигураций 9](#_Toc27480364)

[3. ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ПРОЕКТА 14](#_Toc27480365)

[3.1 Расчет времени создания проекта 14](#_Toc27480366)

[3.2 Расчет себестоимости программного продукта 15](#_Toc27480367)

[3.3 Расчет экономии за счет увеличения производительности труда 16](#_Toc27480368)

[4. ЗАКЛЮЧЕНИЕ 18](#_Toc27480369)

[5. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 19](#_Toc27480370)

[ПРИЛОЖЕНИЕ А ВЗАИМОДЕЙСТВИЕ ЭЛЕМЕНТОВ КОНФИГУРАЦИЙ 20](#_Toc27480371)

[ПРИЛОЖЕНИЕ Б КОД КОНФИГУРАЦИИ СЕРВЕРНОГО ПРИЛОЖЕНИЯ 22](#_Toc27480372)

[ПРИЛОЖЕНИЕ В КОД КОНФИГУРАЦИИ МОБИЛЬНОГО ПРИЛОЖЕНИЯ 27](#_Toc27480373)

# ВВЕДЕНИЕ

По данным Росстата, доля малого и среднего бизнеса в российской экономике составляет 21,9% [1]. Предприятия по изготовлению рекламной продукции (Рекламные агентства) относятся к данной категории. Для стабильной работы и управления предприятием необходимо множество информационных систем, взаимодействующих друг с другом. Однако создание и содержание таких систем может быть очень дорогостоящим, что будет не выгодно для таких предприятий.

Для удешевления и упрощения процесса управления рекламным агентством разумно создать расширяемую систему на базе программного продукта 1С: Предприятие. Данный программный продукт предоставляет возможность для разработки информационной системы с функционалом управления предприятия и ведения бухгалтерии в различном виде (Web-конфигурация, мобильная конфигурация (мобильное приложение) и стандартная конфигурация для компьютера).

В рамках данной работы рассмотрена разработка мобильной конфигурации для возможности взаимодействовать с системой не только за компьютером, но и с использованием мобильного устройства.

# 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## **1.1 Описание предметной области**

Барбершоп в своём классическом варианте - это парикмахерская для мужчин. Присутствие другой половины пола в таких местах исключено по определению. Так как барбершоп является предприятием, предоставляющее людям определённые услуги, то предприятие ведет расписание мастеров и что-нибудь дописать.

На предприятии происходят следующие процессы:

* Ведение расписания мастеров для получения услуг
* Ведение клиентской базы;
* Ведение базы сотрудников
* Может ещё что-то нужно

Исходя из рассмотренных процессов предприятия, в рамках выполнения работы реализован следующий функционал:

1. учет клиентов – персональная информация клиента для связи с ним (ФИО и телефон);
2. учёт мастеров и персонала
3. учёт расписания

**1.2 Цели и задачи разработки**

Цель данного проекта – автоматизация основополагающих процессов работы предприятия для сокращения затрат времени на выполнение посредством внедрения расширяемой информационной системы, включающую серверную и клиентскую конфигурации системы.

Основная задача реализации проекта - настройка соответствий между объектами метаданных конфигураций источника и приемника, а так же настройка соответствий между свойствами объектов (реквизитами объектов и табличных частей). Процесс подготовки файла обмена заканчивается генерацией XML-файла правил и условно разделяется на следующие этапы:

1. Создание двух конфигураций для реализации мобильного и серверного приложений .
2. Создание необходимых объектов конфигураций для реализации проекта (Справочники, Документы, Формы, Отчеты и Регистры).
3. Создание XDTO пакета, для описания структур объектов передачи между сервером и мобильным приложением.
4. Создание Web-сервисов, реализующих взаимодействие между сервером и мобильным приложением.
5. Настройка подключения к Web-сервисам мобильного клиента.

В итоге, необходимые данные должны быть синхронизированы между серверным и мобильным приложениями, для упрощения некоторых процессов предприятия.

## **1.3 Процессы предприятия**

Рассмотрим некоторые процессы предприятия, функционал которых необходимо реализовать. В данный перечень входит:

1. Ведение складского учета материала рекламного агентства
2. Выписка товарно - кассового чека для клиента
3. Расчет финансовых оборотов предприятия

Автоматизация данных процессов в информационной системе позволит не только упростить сам процесс проверки данных, но и обеспечить снижение появление некорректных данных о состоянии финансов предприятии или количества материала.Рассмотрим перечисленные процессы в виде диаграмм последовательности до и после внедрения информационной системы на предприятие.

На рисунке 1 представлен процесс ведения складского учета до внедрения информационной системы на предприятие.

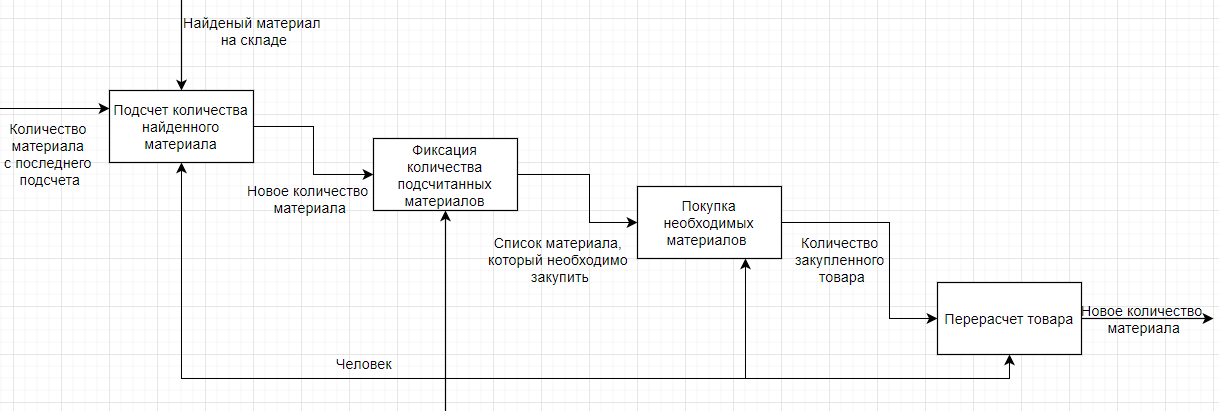


Рисунок 1. Ведение складского учета до внедрения информационной системы

На рисунке 2 представлен процесс ведения складского учета после внедрения информационной системы на предприятие.

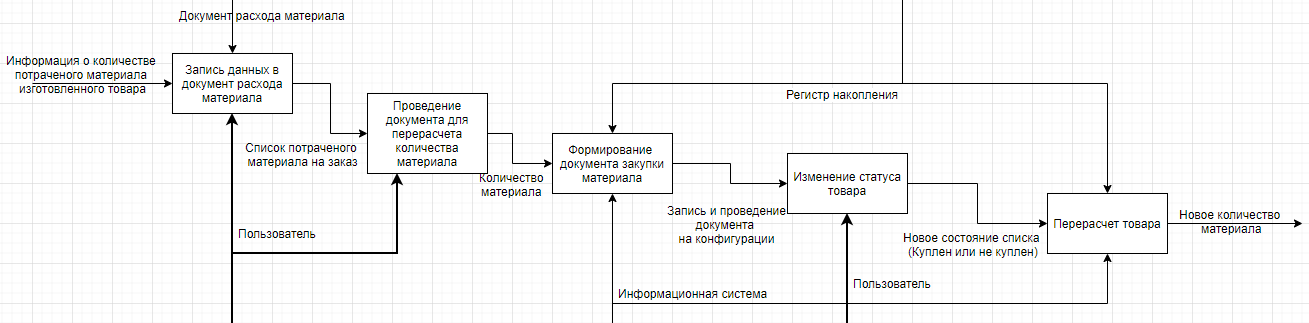


Рисунок 2. Ведение складского учета после внедрения системы

На рисунке 3 представлен процесс ведения учета денежных оборотов предприятия, а так же ведения заказов предприятия до внедрения информационной системы на предприятие.

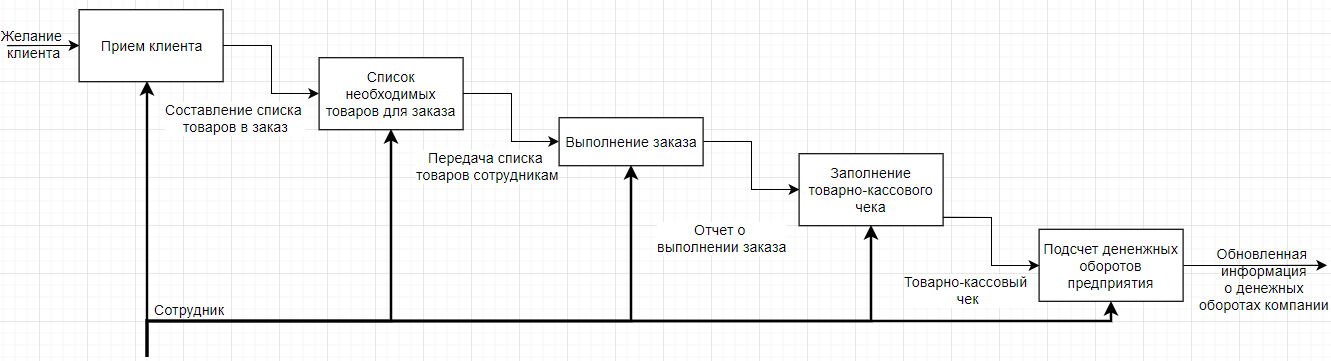


Рисунок 3. Ведение учета денежных средств до внедрения системы

На рисунке 4 представлен процесс ведения учета денежных оборотов предприятия, а так же автоматизация ведения заказов предприятия после внедрения информационной системы на предприятие.

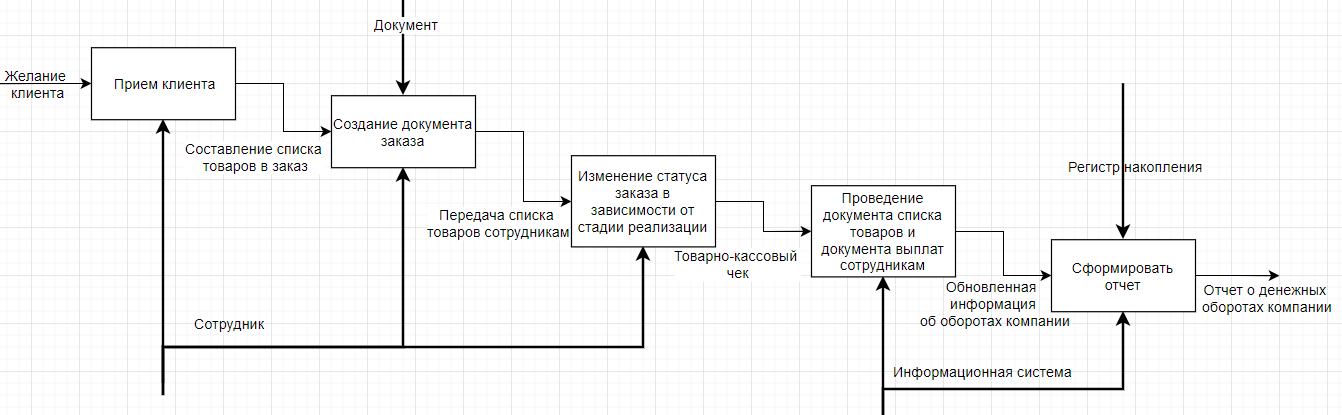


Рисунок 4.Ведение учета денежных средств после внедрения системы

## **1.4 Описание мобильной конфигурации**

Мобильное приложение, установленное на устройстве, представляет собой совокупность мобильной платформы и информационной базы. Информационная база на мобильном устройстве содержит аналог файловой базы данных и мобильное приложение.На рисунке 2 представлена компонентная схема мобильной плапформы.

Основным назначением мобильных приложений является организация выполнения процессов предприятия, удаленно от рабочего места. Также мобильное приложение может быть рассчитано на работу одновременно с несколькими бэк-офисными приложениями. В том числе и с приложениями, созданными не на платформе 1С:Предприятие. На рисунке 5 представлена архитектура взаимодействия мобильного приложения и основной конфигурацией.

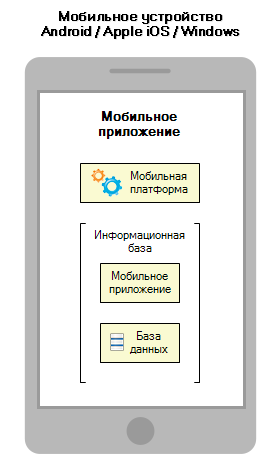


Рисунок 5.Компонентная схема мобильной плапформы

Разработка мобильных приложений ведется так же, как и разработка "обычных" конфигураций 1С, с той лишь разницей, что необходимо учитывать ограничения, которые накладывает мобильная платформа. Для конфигурации, разрабатываемой как мобильное приложение, следует устанавливать свойство «НазначениеИспользования» в значение «МобильноеУстройство». В этом случае система автоматически скроет возможности, недоступные для мобильной платформы, а рабочие инструменты будут настроены на работу именно с тем контекстом встроенного языка, который доступен в мобильной платформе.



Рисунок 6. Архитектура мобильной платформы 1С с одним серверо

Чтобы из конфигурации получить мобильное приложение, готовое для загрузки в магазин приложений, следует выгрузить мобильное приложение в XML файл. Затем, используя "Сборщик мобильных приложений", ,предложенный разработчиками «1С:Предприятие», из XML файла, дистрибутива мобильной платформы и дополнительных файлов собрать дистрибутивы мобильного приложения для операционной системы Android, iOS и Windows.

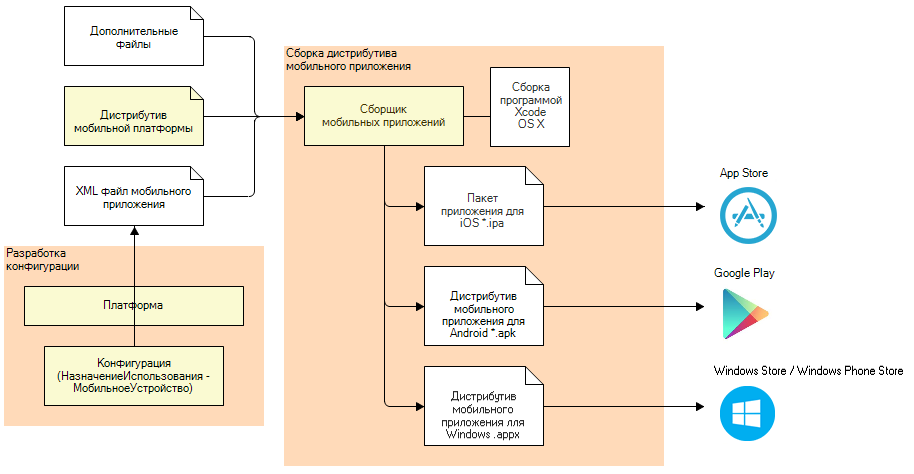


Рисунок 7. Формирование мобильного приложения 1С в зависимости от конечной платформ

# 2 РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ

## **2.1 Составляющие информационной системы**

Информационная система для рекламного агентства состоит их 2 конфигурацийЖ Серверная конфигурация, в которой реализована все необходимые процессы работы предприятия, и мобильной конфигурации, между которыми настроен обмен данными через web-сервисы. Перечень необходимых элементов представлен в таблице 1

Таблица 1. Элементы конфигураций системы

|  |  |  |
| --- | --- | --- |
| **Элемент** | **Серверная конфигурация** | **Мобильная конфигурация** |
| Справочники | Справочник «Материал»  Справочник «Сотрудники»  Справочник «Товар»  Справочник «Статус Заказа»  Справочник «Клиенты» | |
| Документы | Документ «Закупка материала»  Документ «Расход материала»  Документ «Заказ»  Документ «Выплаты Сотрудникам» | Документ «Закупка материала»  Документ «Заказ»  Документ «Выплаты Сотрудникам» |
| Регистры накопления | «Материал»  «Денежный оборот» | - |
| Регистры сведений | - | «Зарегистрированные объекты» |
| Отчеты | «Остатки материала»  «Денежный оборот» | Нет в данной версии конфигурации |
| Взаимосвязь конфигураций | XDTO-пакет  Web-сервис | WS-ссылка |

Диаграммы взаимодействий элементов конфигураций представлены приложении А.

## **2.2. Описание взаимодействия конфигураций**

Для взаимодействия мобильной и серверной конфигураций были реализованы следующие шаги:

1. Установлен AppacheServer для публикаций конфигураций информационной системы. На рисунках 8 и 9 представлены настройки публикаций конфигураций системы

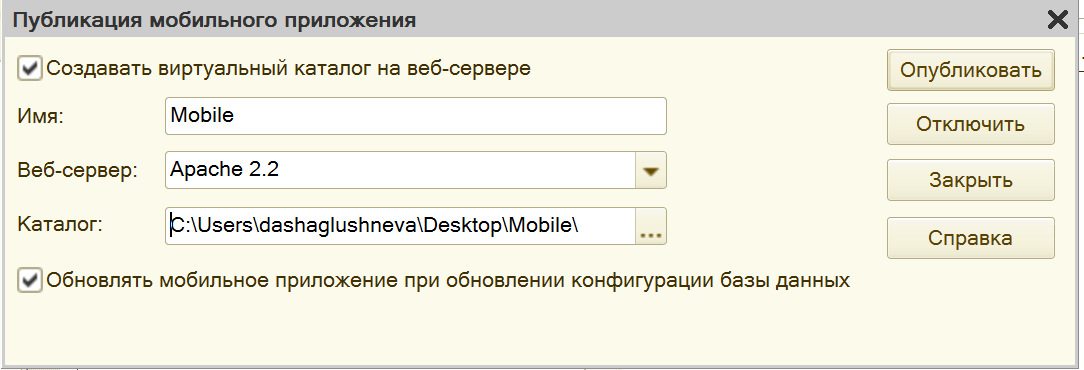


Рисунок 8.Настройки публикации мобильногог приложения

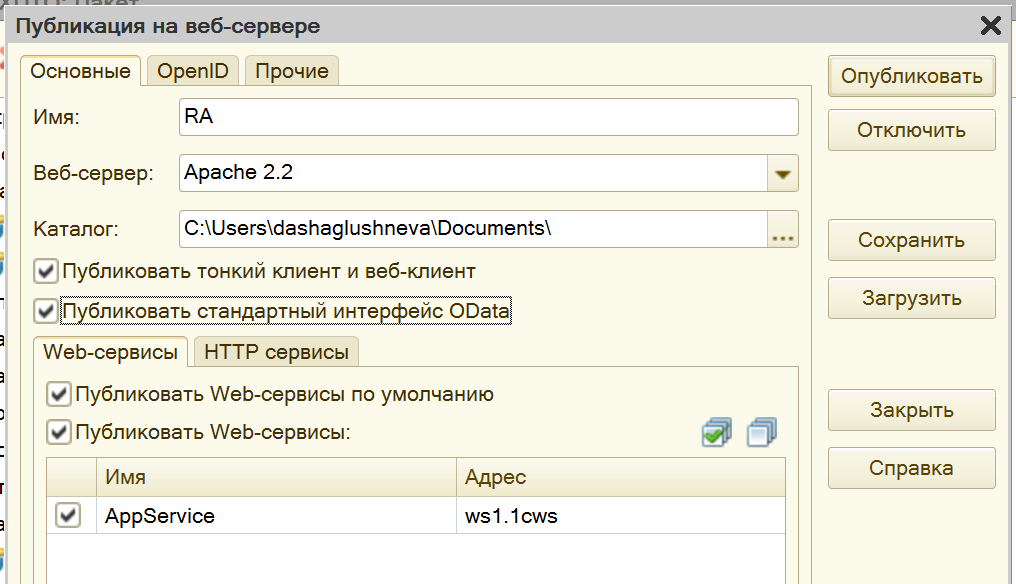


Рисунок 9. Настройка публикации серверной конфигураци

1. Разработан XDTO-пакет, описывающий структуру данных для взаимодействия между конфигурациями. Структура пакеда данной системы представлен на рисунке 10.

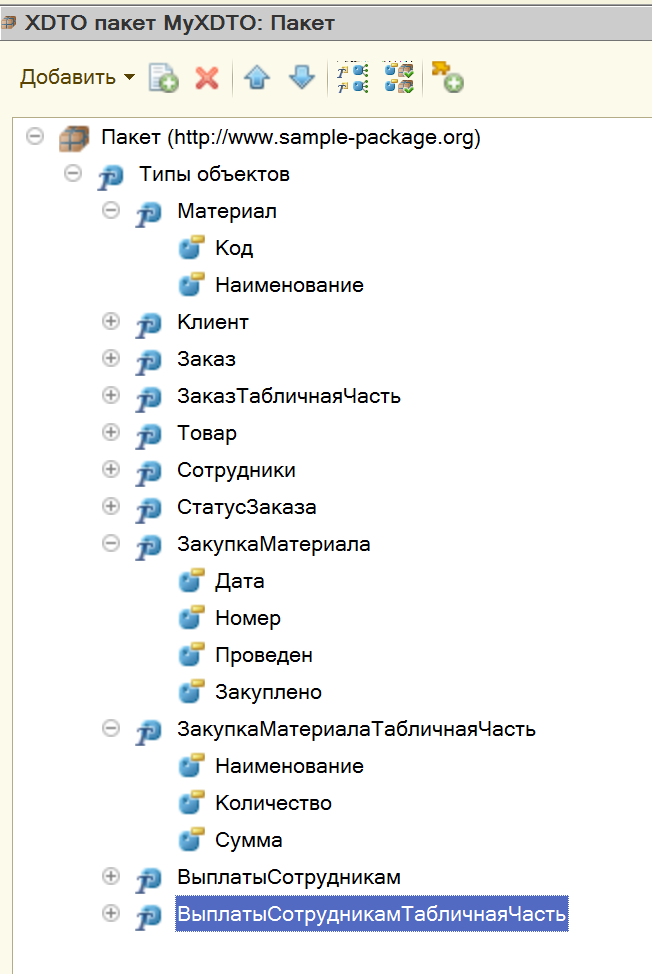


Рисунок 10. Структура XDTO-пакета

1. Написанны web-сервисы для реализации взаимодействия мобильной конфигурации с серверной. На рисунке 11 представлены наименование сервисов и передаваемые параметры. Полный код сервисов представлен в приложении

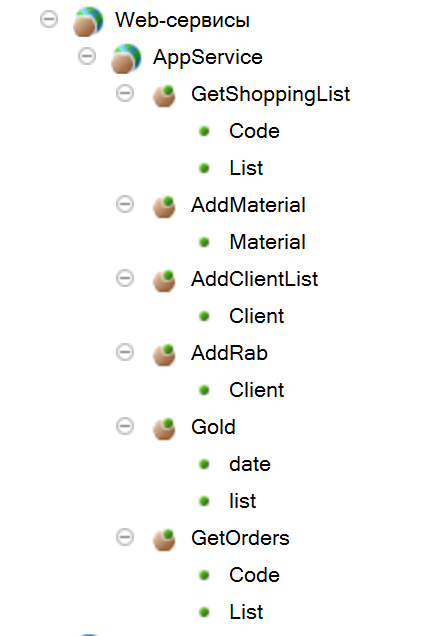


Рисунок 11. Web-сервисы для взаимодействия конфигураций

1. На мобильной конфигурации настроенного подключение к опубликованной серверной конфигурации. На рисунке 12 представлен вариант подключения к развернутой серверной системе в рамках данной разработки

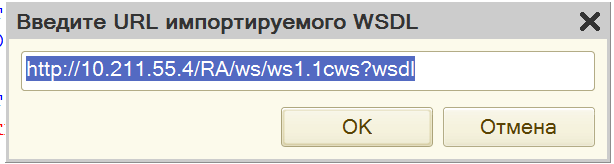


Рисунок 12. Подключение к web-сервисам с мобильной конфигурации

1. Для каждого элемента конфигурации был написан код подключения и взаимодействия с определенным сервисом. Обработка объектов зависела от структуры XDTO-пакета и самого элемента конфигурации. Процесс отправки данных с мобильного приложения на серверное приложение представлен на рисунках 13 - 15. Полный код конфигурации представлены в приложении В

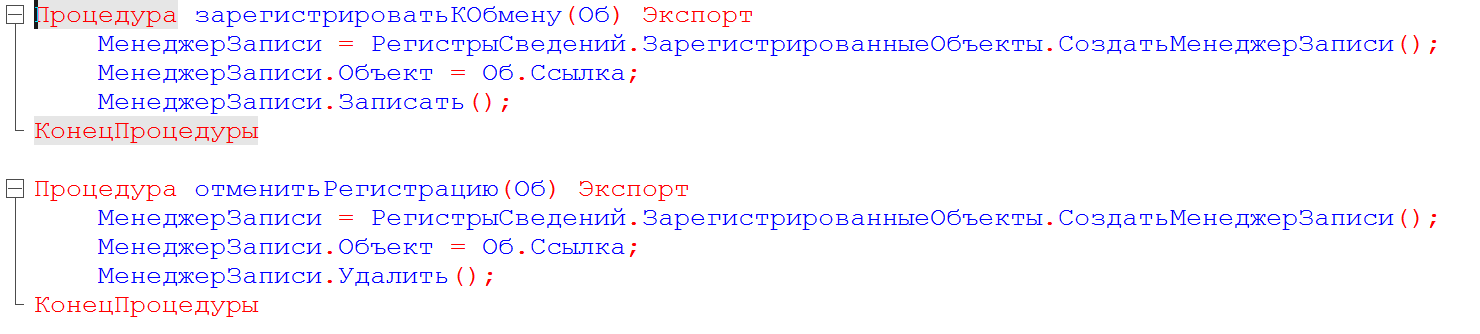


Рисунок 13. Код модуля регистрации изменений данных на мобильном клиенте

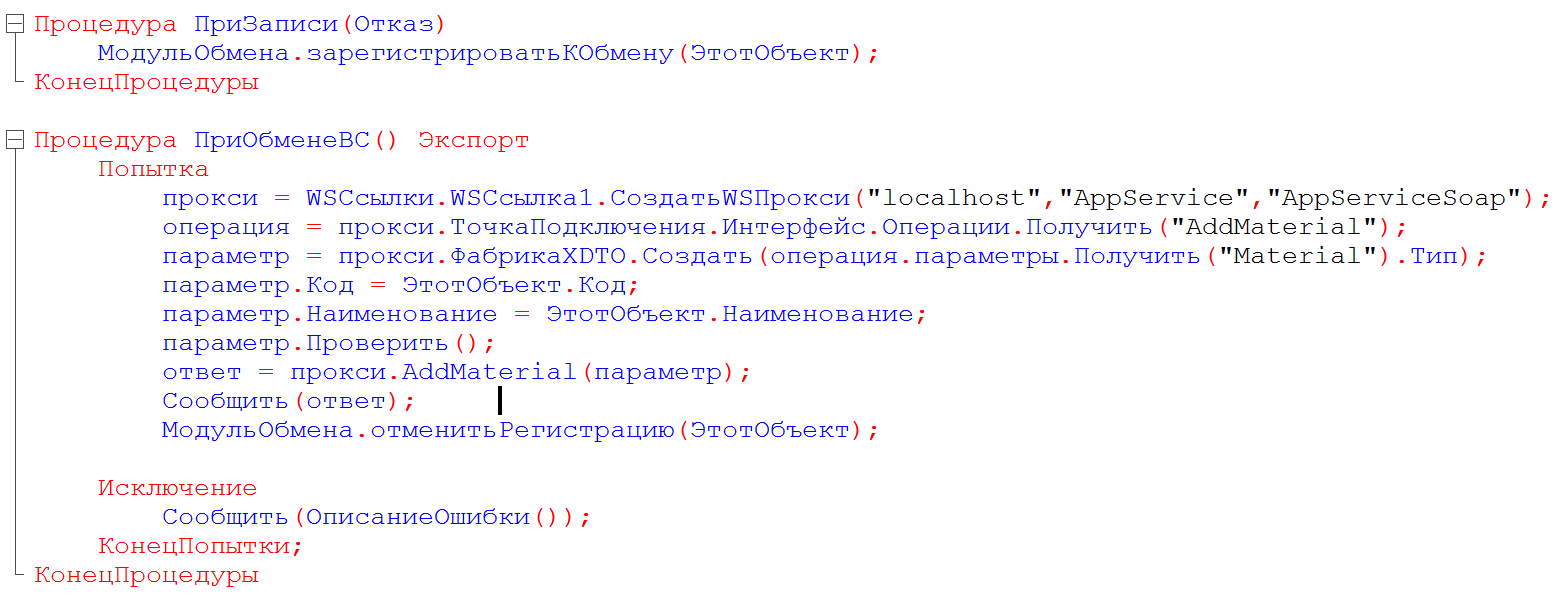


Рисунок 14. Код модуля объекта (Справочник "Материал")

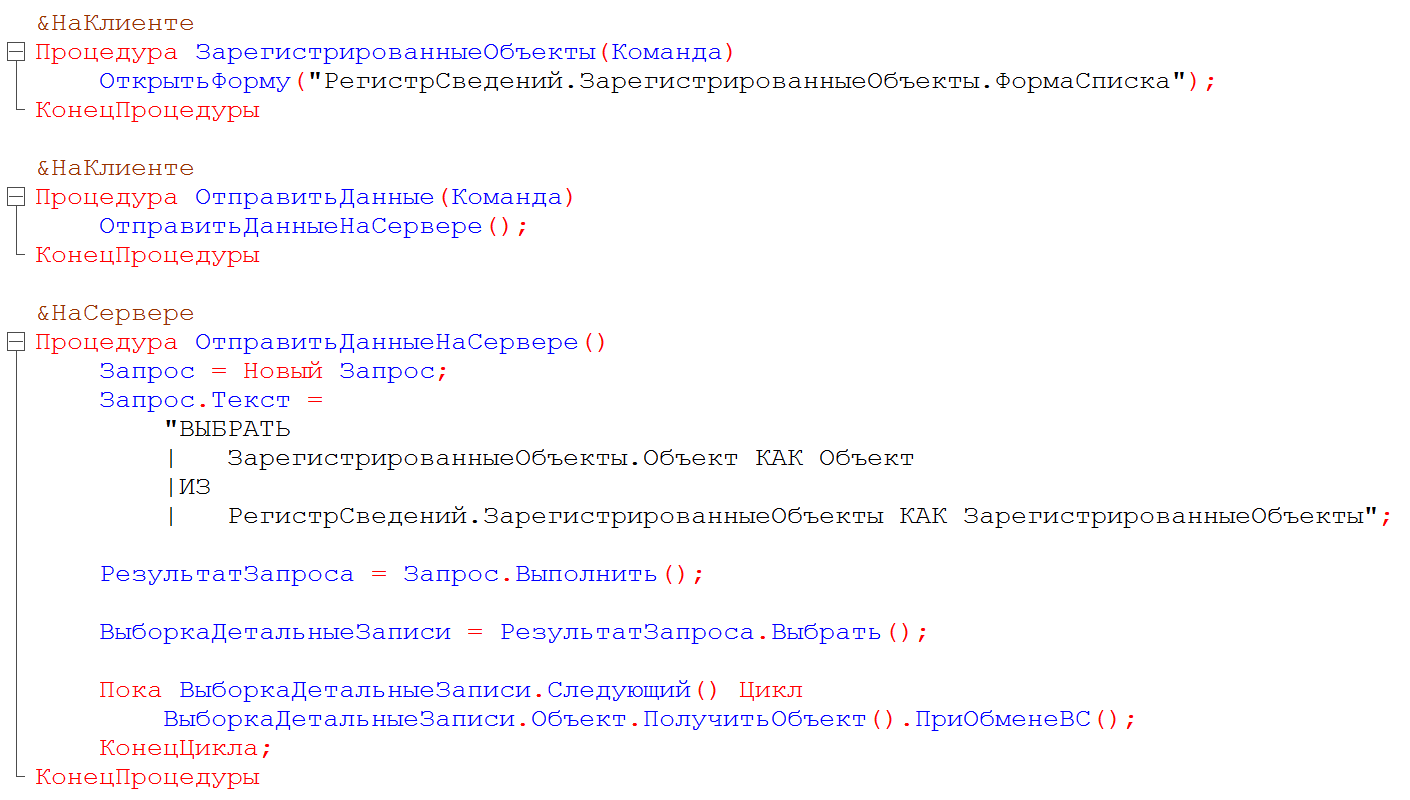


Рисунок 15. Код команд, для отправки данных

# 3. ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ПРОЕКТА

Главный экономический эффект от внедрения средств автоматизации заключается в улучшении экономических и хозяйственных показателей работы предприятия, в первую очередь за счет повышения оперативности управления и снижения трудозатрат на реализацию процесса управления, то есть сокращения расходов на управление. Для большинства предприятий экономический эффект выступает в виде экономии трудовых и финансовых ресурсов, получаемой от:

* снижения трудоемкости расчетов;
* снижение трудозатрат на поиск и подготовку документов;
* экономии на расходных материалах (бумага, дискеты, картриджи);
* сокращения служащих предприятия.

Снижение же трудозатрат на предприятии возможно за счет автоматизации работы с документами, снижения затрат на поиск информации.

Критерием эффективности создания и внедрения новых средств автоматизации является ожидаемый ***экономический эффект***. Он определяется по формуле:

**Э=Эр- Кп, (1)**

где Эр - годовая экономия;

Ен - нормативный коэффициент (Eн=0.15);

Кп - затраты на проектирование и внедрение, включая первоначальную стоимость программы.

**3.1 Расчет времени создания проекта**

В случае если оценивать экономический эффект с учетом всех деталей, то капитальные затраты на проектирование и внедрение рассчитываются с учетом длительности работ на этом этапе. Под проектированием понимается совокупность работ, которые необходимо выполнить, чтобы спроектировать систему, часть системы или поставленную задачу. Под внедрением понимается комплекс работ по вводу в промышленную эксплуатацию системы с возможными ее доработками.

Для расчета затрат на разработку необходимо определить продолжительность каждой работы, начиная с составления технического задания и заканчивая оформлением документов. Продолжительность работ рассчитаны их на основании экспертных оценок по формуле:

Т0=(3\*Тmin+2\*Тmax)/5 (3)

где Т0 - ожидаемая продолжительность работ;

Тmin и Тmax ~ соответственно наименьшая и наибольшая по мнению эксперта длительность работы.

Данные расчетов ожидаемой длительности работ на 1 человека приведены в таблице 2. Данные указаны в днях.

Таблица 2.Таблица длительности работ

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование работ | Длительность работ, дней | | |
| минимум | максимум | ожид. |
| Анализ предметной области | 3 | 5 | 4 |
| Разработка технического задания на проект | 3 | 7 | 5 |
| Проектирование информационной системы | 7 | 14 | 10 |
| Разработка информационной системы | 20 | 35 | 26 |
| Тестирование и отладка программы | 7 | 14 | 10 |
| Оформление документации на информационную систему | 3 | 7 | 5 |
| Ввод в эксплуатацию | 3 | 7 | 5 |
| Итого | 65 (520 часа) | | |

3.2 Расчет себестоимости программного продукта

Себестоимость проекта *Спр* определяется суммированием следующих статей затрат:

– оплата труда разработчика программного продукта;

– техническое обеспечение;

– прочие затраты;

– общехозяйственные расходы.

Средняя заработная плата разработчика 1С по Владимиру (руб./час) составляет 200 руб [6]

*Расчет затрат на техническое обеспечения проекта*

Для разработки программного обеспечения необходимы технические средства:

˗ Монитор 1 шт. - 12000 руб.

˗ системный блок 1 шт. - 9000 руб.

˗ клавиатура 1 шт. - 400 руб.

˗ мышь 1 шт. - 200 руб.

˗ принтер- шт. - 5000 руб.

˗ Windows 10 (лицензионный) - шт. - 5000 руб.

˗ 1С:Предприятие (лицензия) - шт. - 5000 руб.

˗ Лицензия разработчика Android - шт. - 3000 руб.

Итого затрат на техническое и программное обеспечение *С*оф - 34600 руб.

*Расчет затрат на текущий ремонт*

Затраты на текущий и профилактический ремонт принимаются равными 5% от стоимости ЭВМ: *ЗТР* =0,05-*SБАЛ = 0,05\*34600=1730*

Расчитываю себестоимость разработки данного проекта

230\*520+34600+1730 = 140330

**3.3 Расчет экономии за счет увеличения производительности труда**

Если пользователь при экономии i- вида с применением программы экономит Ti, часов, то повышение производительности труда Pi (в %) определяется по формуле:

где Fi - время, которое планировалось пользователем для выполнения работы i-вида до внедрения программы (час.).

Таблица 3.Таблица работ пользователя рекламного агентства

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Вид работ | До автоматизации, мин Fi | Экономия времени, мин.DT | Повышение про­изводительности труда Рi (в %) |
| 1. | Выписка товарно-кассового чека | 15 | 13 | 650 |
| 2. | Составление списка покупок необходимого материала | 60 | 50 | 500 |

Экономия, связанная с повышением производительности труда пользователя Р определим по формуле:

где Zп - среднегодовая заработная плата пользователя.

Экономия, связанная с повышением производительности труда пользователя составляет: P = 30000 \* 11,5 = 345000 руб.

В итоге получаем следующую ожидаемую экономическую эффективность:

Э = 345000 – 140330 = 204670 руб.

Даже при приблизительном расчете экономическая эффективность от внедрения программного средства получилась значительной. Такой она получилась за счет увеличения производительности труда сотрудника.

Соответственно потратив 140330 рублей получаем экономию за год в 204670 рублей.

# 4. ЗАКЛЮЧЕНИЕ

В ходе работы была разработана информационная система для рекламного агентства. Для удобства работы с системой были созданы различные методы для взаимодействия разных пользователей с системой.

Разработанная система является актуальной на сегодняшний день и имеет большую практическую значимость. Она может помочь в работе сотрудников рекламного агентства. В ней легко организовывать поиск, оформлять заявки, вести справочник, так как предусмотрена эффективная организация, минимизирующая время доступа и поиска.

В результате выполнения курсового проекта была создана информационная система для автоматизации работы процессов работы рекламного агентства.

В данной информационной системе присутствуют:

* справочники;
* документы;
* регистры накоплений;
* отчеты;
* Web-сервис для осуществления обмена данными.

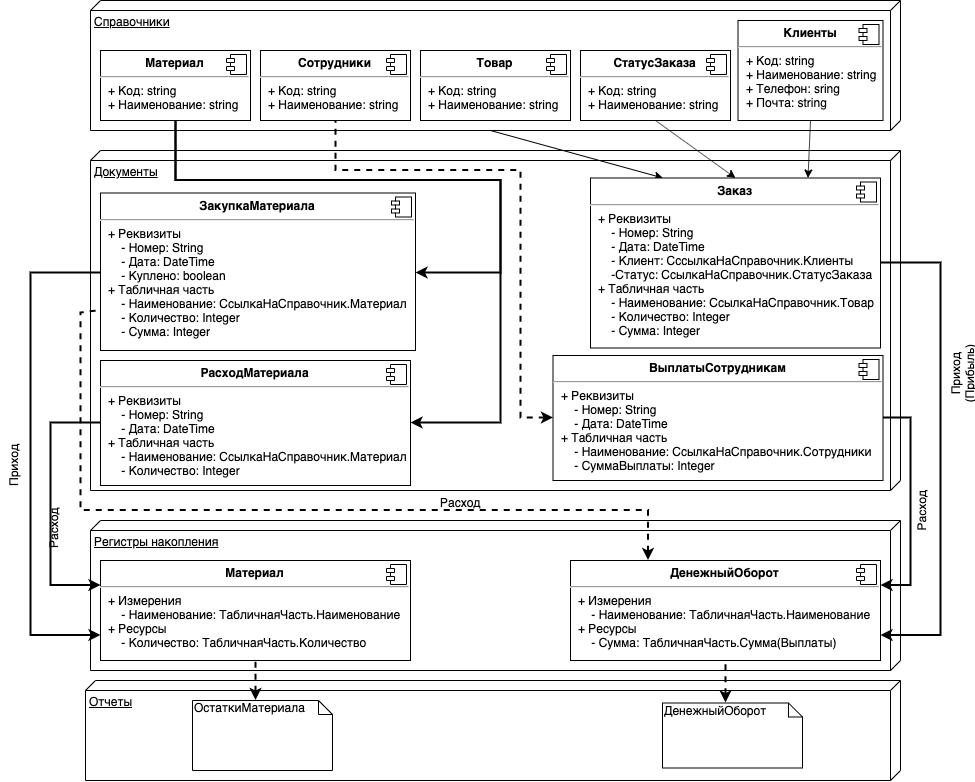
Созданная система позволяет повысить эффективность работы предприятия за счет уменьшения времени, потраченного на заполнение документов и отчетов. Себестоимость программного продукта составила 140330 рублей. Экономия предприятия после внедрения информационной системы составляет 204670 рублей.

# 5. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

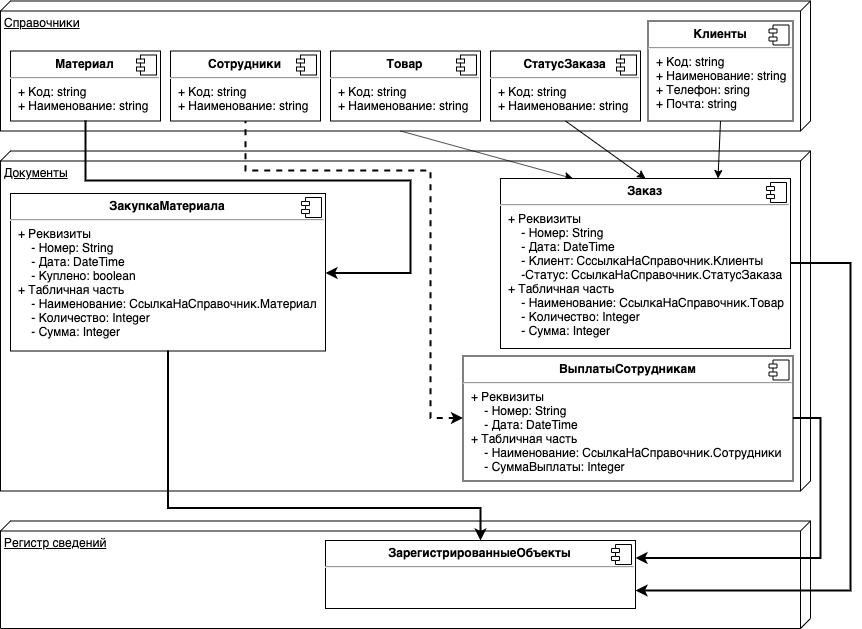
1. Экономические данные Росстата (Электронный ресурс). Режим доступа: https://www.rbc.ru/economics/05/02/2019/5c5948c59a794758389cfdf7 (Дата обращения: 12.12.19)

# ПРИЛОЖЕНИЕ А ВЗАИМОДЕЙСТВИЕ ЭЛЕМЕНТОВ КОНФИГУРАЦИЙ

А.1. Конфигурация серверного приложения



А.2. Конфигурация мобильного приложения



# ПРИЛОЖЕНИЕ Б КОД КОНФИГУРАЦИИ СЕРВЕРНОГО ПРИЛОЖЕНИЯ

Б.1. Код Web-сервисов

Функция AddClientList(Client)

Попытка

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| объекты.Ссылка КАК Ссылка

|ИЗ

| Справочник.Клиент КАК Объекты

|ГДЕ

| Объекты.Код = &Код";

Запрос.УстановитьПараметр("Код", Client.Код);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

объекты = ВыборкаДетальныеЗаписи.Ссылка.ПолучитьОбъект();

Иначе

объекты = Справочники.Клиент.СоздатьЭлемент();

объекты.Код = Client.Код;

КонецЕсли;

объекты.Наименование = Client.Наименование;

объекты.Телефон = Client.Телефон;

объекты.Почта = Client.Почта;

объекты.Записать();

Исключение

Возврат ОписаниеОшибки();

КонецПопытки;

Возврат "Успешно";

КонецФункции

Функция AddMaterial(Material)

Попытка

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| объекты.Ссылка КАК Ссылка

|ИЗ

| Справочник.Материал КАК Объекты

|ГДЕ

| Объекты.Код = &Код";

Запрос.УстановитьПараметр("Код", Material.Код);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

объекты = ВыборкаДетальныеЗаписи.Ссылка.ПолучитьОбъект();

Иначе

объекты = Справочники.Материал.СоздатьЭлемент();

объекты.Код = Material.Код;

КонецЕсли;

объекты.Наименование = Material.Наименование;

объекты.Записать();

Исключение

Возврат ОписаниеОшибки();

КонецПопытки;

Возврат "Успешно";

КонецФункции

Функция GetShoppingList(Code, List)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ЗакупкаМатериала.Ссылка КАК Ссылка

|ИЗ

| Документ.ЗакупкаМатериала КАК ЗакупкаМатериала

|ГДЕ

| ЗакупкаМатериала.Номер = &Номер";

Запрос.УстановитьПараметр("Номер", Code.Номер);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

лист1 = ВыборкаДетальныеЗаписи.Ссылка.ПолучитьОбъект();

СтатусДокумента = "найден";

Иначе

лист1 = Документы.ЗакупкаМатериала.СоздатьДокумент();

лист1.Номер = Code.Номер;

СтатусДокумента = "создан";

КонецЕсли;

лист1.Дата = Code.Дата;

лист1.Закуплено = Code.Закуплено;

ТЗ = Новый ТаблицаЗначений;

ТЗ.Колонки.Добавить("Наименование");

ТЗ.Колонки.Добавить("Количество");

ТЗ.Колонки.Добавить("Сумма");

Для ит = 0 по List.Наименование.Количество()-1 Цикл // Схождение кол-ва строчек

стр = тз.Добавить();

стр.Наименование = Справочники.Материал.НайтиПоКоду(List.Наименование[ит]);

стр.Количество = List.Количество[ит];

стр.Сумма = List.Сумма[ит];

КонецЦикла;

Для каждого эл из тз Цикл

стр = лист1.Сотрудники.Добавить();

стр.Наименование = эл.Наименование;

стр.Количество = эл.Количество;

стр.СуммаВыплаты = эл.СуммаВыплаты;

КонецЦикла;

Лист1.Записать(РежимЗаписиДокумента.Запись);

СтатусДокумента = "записан";

Если Code.Проведен Тогда

Лист1.Записать(РежимЗаписиДокумента.Проведение);

СтатусДокумента = "проведен";

Иначе

Лист1.Записать(РежимЗаписиДокумента.ОтменаПроведения);

СтатусДокумента = "не проведен";

КонецЕсли;

Возврат СтатусДокумента;

КонецФункции

Функция AddRab(Client)

Попытка

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Сотрудники.Ссылка КАК Ссылка

|ИЗ

| Справочник.Сотрудники КАК Сотрудники

|ГДЕ

| Сотрудники.Код = &Код";

Запрос.УстановитьПараметр("Код", Client.Код);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

Сотрудники = ВыборкаДетальныеЗаписи.Ссылка.ПолучитьОбъект();

Иначе

Сотрудники = Справочники.Сотрудники.СоздатьЭлемент();

Сотрудники.Код = Client.Код;

КонецЕсли;

Сотрудники.Наименование = Client.Наименование;

Сотрудники.Записать();

Исключение

Возврат ОписаниеОшибки();

КонецПопытки;

Возврат "Успешно";

КонецФункции

Функция Gold(date, list)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ВыплатыСотрудникам.Ссылка КАК Ссылка

|ИЗ

| Документ.ВыплатыСотрудникам КАК ВыплатыСотрудникам

|ГДЕ

| ВыплатыСотрудникам.Номер = &Номер";

Запрос.УстановитьПараметр("Номер", date.Номер);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

лист1 = ВыборкаДетальныеЗаписи.Ссылка.ПолучитьОбъект();

СтатусДокумента = "найден";

Иначе

лист1 = Документы.ВыплатыСотрудникам.СоздатьДокумент();

лист1.Номер = date.Номер;

СтатусДокумента = "создан";

КонецЕсли;

лист1.Дата = date.Дата;

ТЗ = Новый ТаблицаЗначений;

ТЗ.Колонки.Добавить("Наименование");

ТЗ.Колонки.Добавить("СуммаВыплаты");

Для ит = 0 по list.Наименование.Количество()-1 Цикл // Схождение кол-ва строчек

стр = тз.Добавить();

стр.Наименование = Справочники.Сотрудники.НайтиПоКоду(list.Наименование[ит]);

стр.СуммаВыплаты = list.СуммаВыплаты[ит];

КонецЦикла;

Для каждого эл из тз Цикл

стр = лист1.Сотрудники.Добавить();

стр.Наименование = эл.Наименование;

стр.СуммаВыплаты = эл.СуммаВыплаты;

КонецЦикла;

Лист1.Записать(РежимЗаписиДокумента.Запись);

СтатусДокумента = "записан";

Если date.Проведен Тогда

Лист1.Записать(РежимЗаписиДокумента.Проведение);

СтатусДокумента = "проведен";

Иначе

Лист1.Записать(РежимЗаписиДокумента.ОтменаПроведения);

СтатусДокумента = "не проведен";

КонецЕсли;

Возврат СтатусДокумента;

КонецФункции

Функция GetOrders(Code, List)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Заказ.Ссылка КАК Ссылка

|ИЗ

| Документ.Заказ КАК Заказ

|ГДЕ

| Заказ.Номер = &Номер";

Запрос.УстановитьПараметр("Номер", Code.Номер);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

лист1 = ВыборкаДетальныеЗаписи.Ссылка.ПолучитьОбъект();

СтатусДокумента = "найден";

Иначе

лист1 = Документы.Заказ.СоздатьДокумент();

лист1.Номер = Code.Номер;

СтатусДокумента = "создан";

КонецЕсли;

лист1.Дата = Code.Дата;

лист1.Клиент = Справочники.Клиент.НайтиПоКоду(Code.Клиент);

лист1.Статус = Справочники.СтатусЗаказа.НайтиПоКоду(Code.Статус);

ТЗ = Новый ТаблицаЗначений;

ТЗ.Колонки.Добавить("Товар");

ТЗ.Колонки.Добавить("Количество");

ТЗ.Колонки.Добавить("Сумма");

Для ит = 0 по List.Наименование.Количество()-1 Цикл // Схождение кол-ва строчек

стр = тз.Добавить();

стр.Товар = Справочники.Товар.НайтиПоКоду(List.Товар[ит]);

стр.Количество = List.Количество[ит];

стр.Сумма = List.Сумма[ит];

КонецЦикла;

Для каждого эл из тз Цикл

стр = лист1.Товары.Добавить();

стр.Товар = эл.Товар;

стр.Количество = эл.Количество;

стр.Сумма = эл.Сумма;

КонецЦикла;

Лист1.Записать(РежимЗаписиДокумента.Запись);

СтатусДокумента = "записан";

Если Code.Проведен Тогда

Лист1.Записать(РежимЗаписиДокумента.Проведение);

СтатусДокумента = "проведен";

Иначе

Лист1.Записать(РежимЗаписиДокумента.ОтменаПроведения);

СтатусДокумента = "не проведен";

КонецЕсли;

Возврат СтатусДокумента;

КонецФункции

# ПРИЛОЖЕНИЕ В КОД КОНФИГУРАЦИИ МОБИЛЬНОГО ПРИЛОЖЕНИЯ

Б.1. Код общего модуля

Процедура зарегистрироватьКОбмену(Об) Экспорт

МенеджерЗаписи = РегистрыСведений.ЗарегистрированныеОбъекты.СоздатьМенеджерЗаписи();

МенеджерЗаписи.Объект = Об.Ссылка;

МенеджерЗаписи.Записать();

КонецПроцедуры

Процедура отменитьРегистрацию(Об) Экспорт

МенеджерЗаписи = РегистрыСведений.ЗарегистрированныеОбъекты.СоздатьМенеджерЗаписи();

МенеджерЗаписи.Объект = Об.Ссылка;

МенеджерЗаписи.Удалить();

КонецПроцедуры

Б.2. Справочник «Материал»

Код модуля объекта

Процедура ПриЗаписи(Отказ)

МодульОбмена.зарегистрироватьКОбмену(ЭтотОбъект);

КонецПроцедуры

Процедура ПриОбменеВС() Экспорт

Попытка

прокси = WSСсылки.WSСсылка1.СоздатьWSПрокси("localhost","AppService","AppServiceSoap");

операция = прокси.ТочкаПодключения.Интерфейс.Операции.Получить("AddMaterial");

параметр = прокси.ФабрикаXDTO.Создать(операция.параметры.Получить("Material").Тип); параметр.Код = ЭтотОбъект.Код;

параметр.Наименование = ЭтотОбъект.Наименование;

параметр.Проверить();

ответ = прокси.AddMaterial(параметр);

Сообщить(ответ);

МодульОбмена.отменитьРегистрацию(ЭтотОбъект);

Исключение

Сообщить(ОписаниеОшибки());

КонецПопытки;

КонецПроцедуры

Код формы списка

&НаКлиенте

Процедура ЗарегистрированныеОбъекты(Команда)

ОткрытьФорму("РегистрСведений.ЗарегистрированныеОбъекты.ФормаСписка");

КонецПроцедуры

&НаКлиенте

Процедура ОтправитьДанные(Команда)

ОтправитьДанныеНаСервере();

КонецПроцедуры

&НаСервере

Процедура ОтправитьДанныеНаСервере()

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ЗарегистрированныеОбъекты.Объект КАК Объект

|ИЗ

| РегистрСведений.ЗарегистрированныеОбъекты КАК ЗарегистрированныеОбъекты";

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

ВыборкаДетальныеЗаписи.Объект.ПолучитьОбъект().ПриОбменеВС();

КонецЦикла;

КонецПроцедуры