

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**  
(НИУ «БелГУ»)

ИНСТИТУТ ИНЖЕНЕРНЫХ И ЦИФРОВЫХ ТЕХНОЛОГИЙ  
КАФЕДРА ИНФОРМАЦИОННЫХ И РОБОТОТЕХНИЧЕСКИХ СИСТЕМ

**Разработка протоколов обработки нормативно-справочной информации и  
выгрузки реестров счетов для медицинской организации Московской  
области**

Выпускная квалификационная работа  
обучающегося по направлению подготовки 09.03.02  
Информационные системы и технологии  
очной формы обучения, группы 12001509  
Рышковой Анастасии Николаевны

Научный руководитель  
доцент  
Игрунова С.В.

## РЕФЕРАТ

Разработка протоколов обработки нормативно-справочной информации и выгрузки реестров счетов для медицинской организации Московской области – Рышковой Анастасии Николаевны выпускная квалификационная работа бакалавра, Белгород, Белгородский государственный национальный исследовательский университет (НИУ «БелГУ»), количество страниц 64, включая приложения 81, количество рисунков 35, включая приложения 42, количество таблиц 2, количество использованных источников 35.

**КЛЮЧЕВЫЕ СЛОВА:** информационная система, база данных, протокол, разработка, анализ.

**ОБЪЕКТ ИССЛЕДОВАНИЯ:** процессы сбора и обработки данных, с целью выполнения анализа состояния нормативно-справочной информации и подготовки пакетов данных для выгрузки реестров счетов в медицинских организациях Московской области на основании организационно-технического регламента.

**ПРЕДМЕТ ИССЛЕДОВАНИЯ:** процесс актуализации НСИ и корректно выставленные на оплату пакеты реестров счетов.

**ЦЕЛЬ РАБОТЫ:** усовершенствование работы отдела взаиморасчетов при выгрузке реестров счетов, за счет автоматизации обработки пакета НСИ.

**ЗАДАЧИ ИССЛЕДОВАНИЯ:** изучение предметной области; выявление недостатков существующего механизма обработки НСИ и выгрузки реестров счетов; обоснование выбора основных проектных решений и необходимости создания протоколов обработки НСИ и выгрузки реестров счетов; разработка протоколов обработки НСИ и выгрузки реестров счетов для медицинской организации Московской области; тестирование разработанного протокола выгрузки реестров счетов в тестовый период и обработки НСИ.

**МЕТОДЫ ИССЛЕДОВАНИЯ:** анализ деятельности организации с целью обоснования необходимости разработки протоколов, метод моделирования бизнес-процессов, методы разработки протоколов.

**ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ:** в результате работы были разработаны протоколы обработки нормативно-справочной информации и выгрузки реестров счетов для медицинской организации Московской области, которые позволяют обрабатывать НСИ, путем автоматизированного внесения и изменений данных и выгружать реестры счетов в соответствии с организационно-техническим регламентом.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	4
1 Аналитическая часть .....	6
1.1 Характеристика предметной области .....	6
1.1.1 Характеристика предприятия .....	6
1.1.2 Характеристика отдела взаиморасчетов.....	8
1.2 Обоснование необходимости создания протоколов обработки НСИ и выгрузки реестров счетов.....	9
1.3 Постановка задачи .....	18
2 Обоснование проектных решений .....	23
2.1 Обоснование проектных решений по техническому обеспечению.....	23
2.2 Обоснование проектных решений по информационному обеспечению	24
2.3 Обоснование проектных решений по программному обеспечению .....	28
2.4 Обоснование проектных решений по технологическому обеспечению	32
3 Программная реализация проектных решений.....	37
3.1 Разработка функциональной модели .....	37
3.2 Структура базы данных .....	42
3.3 Пример работы программного обеспечения .....	44
3.4 Описание контрольного примера реализации проекта .....	49
ЗАКЛЮЧЕНИЕ .....	61
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	62
ПРИЛОЖЕНИЕ А .....	66
ПРИЛОЖЕНИЕ В .....	69

## ВВЕДЕНИЕ

Ежедневно в медицинские организации за медицинской помощью обращаются граждане, которых необходимо идентифицировать в регистре застрахованных лиц, согласно предоставленным документам. Медицинская организация при этом направляет в территориальный фонд обязательного медицинского страхования Московской области (ТФОМС МО) файлы - реестры обратившихся пациентов. При установлении страховой принадлежности и отсутствии ошибок в форматно-логическом контроле, данные о пациентах отправляются в страховые компании, в следствии чего больницы получают финансирование по оказанной медицинской помощи. Разработка протоколов загрузки нормативно-справочной информации (НСИ) и выгрузки реестров счетов для медицинской организации Московской области (МО) является средством корректного оформления реестров обратившихся пациентов, вследствие чего, медицинские организации получают финансирование по оказанной медицинской помощи, что подтверждает актуальность и важность задачи.

На текущий момент механизма автоматизированной обработки НСИ не существует, и в итоге обработка информации происходит медленно, зачастую сопровождающаяся ошибками, что влечет за собой отказ оплаты медицинской помощи со стороны страховых медицинских компаний (СМО).

Автоматизация этого процесса поможет не только ускорить работу, но и повысить ее качество, а также уменьшить погрешность некорректного оформления медицинской помощи.

Целью выпускной квалификационной работы является усовершенствование работы отдела взаиморасчетов при выгрузке реестров счетов, за счет автоматизации обработки пакета НСИ

Задачи, решаемые в выпускной квалификационной работе:

- изучение предметной области;

- выявление недостатков существующего механизма обработки НСИ и выгрузки реестров счетов;
- обоснование выбора основных проектных решений;
- обоснование необходимости создания протоколов обработки НСИ и выгрузки реестров счетов;
- разработка протоколов обработки нормативно-справочной информации и выгрузки реестров счетов для медицинской организации Московской области;
- тестирование разработанного протокола выгрузки реестров счетов в тестовый период и обработки НСИ.

Для выполнения поставленных целей необходимо хорошо знать принципы взаимодействия медицинских организаций и ТФОМС МО, реализовать проектирование вариант решения, а затем реализовать его.

Выпускная квалификационная работа выполнена на 64 страницах без приложения, и 81 с приложением, содержит 35 рисунков, 2 таблицы, 4 листинга и приложение.

# 1 Аналитическая часть

## 1.1 Характеристика предметной области

### 1.1.1 Характеристика предприятия

Объектом исследования являются процессы сбора и обработки данных, с целью выполнения анализа состояния нормативно-справочной информации (НСИ) и подготовки пакетов данных для выгрузки реестров счетов в медицинских организациях Московской области на основании организационно-технического регламента (ОТР) [15].

Предметом исследования является процесс актуализации НСИ и корректно выставленные на оплату пакеты реестров счетов. Данные пакеты необходимы для прохождения форматно-логического контроля(ФЛК), сверки и идентификации по единому регистру застрахованных(ЕРЗ), в том числе для оплаты медицинской помощи, лицам, зарегистрированным в других субъектах Российской Федерации, разработанный в рамках компании «СофтТраст».

Компания «СофтТраст» – разработчик программного обеспечения. Продукты компании создаются для врачей и их пациентов с 2005 года. Одним из главных продуктов «СофтТраст» является медицинская информационная система «ТМ:МИС» предназначена для комплексной автоматизации деятельности персонала лечебно-профилактических учреждений, включающих стационарные и амбулаторно-поликлинические подразделения. Система может функционировать как централизованно - на уровне центра обработки данных (ЦОД), так и на уровне медицинской организации путем развертывания локального веб-сервера и предоставления доступа авторизованного медицинского персонала других медицинских организаций.

На сегодняшний день модель компании «СофТраст» включает в себя ТрастМед:МИС - комплексная автоматизация деятельности лечебно-профилактического учреждения; ТрастМед: Больничная аптека - Автоматизация назначения лекарственных препаратов пациентам с последующим автоматическим персонифицированным списанием со складов отделений и больничной аптеки; Личный кабинет пациента - веб-сервис для записи на приём к врачу, ведения журнала здоровья, прохождения диспансеризаций и медицинских осмотров; ТрастМед:Поликлиника - Комплексная автоматизация деятельности персонала амбулаторно-поликлинического подразделения; ТрастМед:Стационар - комплексная автоматизация деятельности персонала стационара; ТрастМед: Шаблон - веб-сервис для создания шаблонов медицинских записей.

Данная модель отображена на рисунке 1.

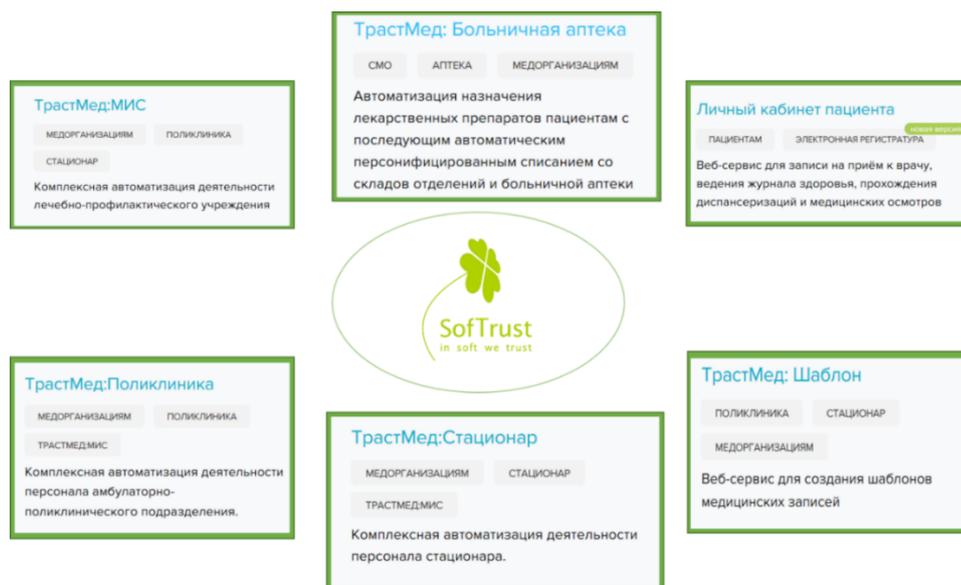


Рисунок 1 - Модель компании «СофТраст»

«СофТраст» заявил о себе в нескольких областях, например, Белгородской, Московской, Орловской, Владимирской и др., а также обслуживает жизни миллионов.

Основная работа происходит в клиентском приложении «ТМ:МИС». Пользователями МИС являются:

- медицинский персонал регистратуры;
- врачи-специалисты поликлиники;
- медицинский персонал приемного покоя;
- медицинский персонал отделений;
- врачи больничных стационаров;
- медицинский персонал аптеки отделений.

### 1.1.2 Характеристика отдела взаиморасчетов

Следует подробно описать деятельность подразделения взаиморасчетов, его структуру, основные выполняемые функции. Ниже представлены задачи и функции, за которые отвечает отдел взаиморасчетов.

Задачами отдела взаиморасчетов являются:

- обеспечение штатного функционирования отчетных форм, разработанных отделом взаиморасчетов;
- обеспечение штатного функционирования протоколов, разработанных отделом взаиморасчетов;
- обеспечение штатного функционирования критериев в экспертизу разработанных отделом взаиморасчетов;
- сопровождение в эксплуатации медицинскими организациями протоколов и отчетных форм.

Отдел взаиморасчетов, в рамках поставленных задач, выполняет следующие функции:

- отвечает за обеспечение штатной эксплуатации протоколов и отчетной, касающиеся сдачи реестров счетов и поддержку деятельности пользователей структурных подразделений других отделов;
- обеспечивает администрирование баз данных, прикладного программного обеспечения и технических средств информационно-технической инфраструктуры управления, устранение нештатных ситуаций, связанных с

функционированием баз данных, прикладного программного обеспечения и технических средств информационно-технической инфраструктуры управления;

– обеспечивает мониторинг новых правил контроля для выгрузки реестров счетов, а также версию пакета НСИ на ТФОМС МО (территориальный фонд обязательного медицинского страхования);

– отвечает за организацию работы по подготовке тестовых сред для проведения предварительных испытаний и опытной эксплуатации информационных систем;

– принимает участие в проведении предварительных испытаний и опытной эксплуатации информационных систем.

## 1.2 Обоснование необходимости создания протоколов обработки НСИ и выгрузки реестров счетов

В первую очередь, необходимо проанализировать эффективность работы отдела взаиморасчетов, выявить недостатки системы на данный момент и исходя из вышеперечисленных данных, найти способы совершенствования системы и автоматизации отдела в целом. Данный анализ позволит оценить ресурсы (информационные, человеческие и другие), используемые при осуществлении рассматриваемых процессов.

До момента создания протоколов загрузки НСИ и выгрузки реестров счетов, автоматизированных механизмов выполнения поставленных задач не имелось.

Ежемесячно, любой процесс изменения справочников начинался с анализа новых пакетов НСИ: выявления новых записей, проверка на соответствие уже имеющихся записей в базе данных и актуальных пакетов НСИ, проверка на наличие действующих записей. Каждый процесс выполнялся вручную [4].

Основные диаграммы были реализованы с помощью методологии IDEF0, эта методология была выбрана, так как при построении диаграмм не было

необходимости детально отображать все тонкости процесса, хотя сама методология IDEF0 – строгая методика [14].

Главные преимущества состоят в следующем:

- возможность сохранения комплексности при декомпозиции;
- наличие детализации, разделении и слиянии потоков данных и информации;
- требование методологий, которые обеспечивают получение моделей процессов в стандартном виде.

Контекстная диаграмма обработки пакетов нормативно-справочной информации представлена на рисунке 2.

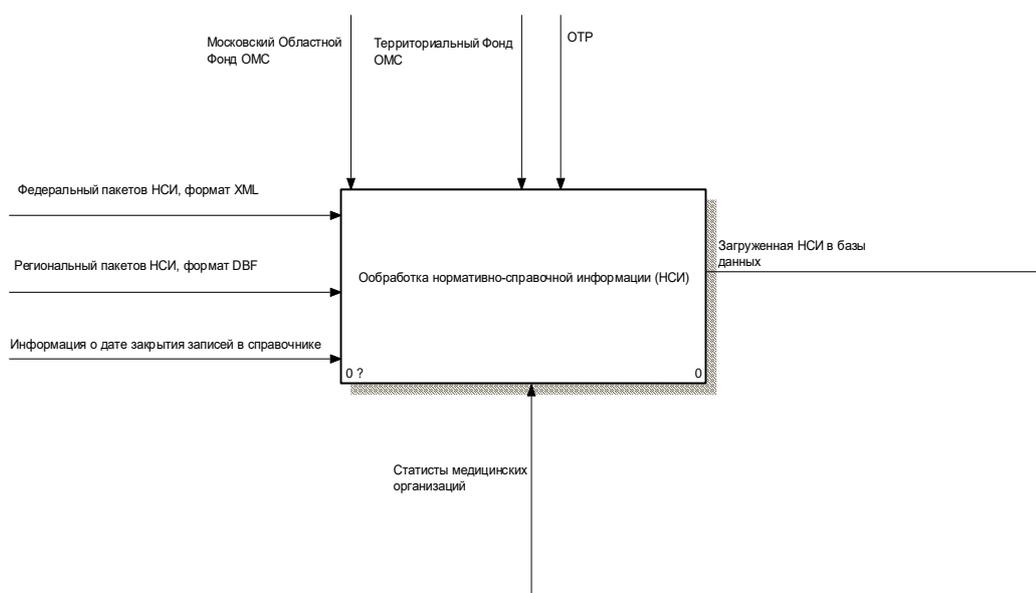


Рисунок 2 - Контекстная диаграмма «Обработка НСИ»

На контекстной диаграмме представлен главный блок «обновление пакетов НСИ». Данный блок обобщает всю рассматриваемую деятельность по обновлению НСИ.

В разработанной модели по методологии IDEF0 присутствуют три входа и один. Вся выполняемая деятельность должна контролироваться какой-то нормативной информацией, контроль на модели отображается верхними стрелками. В данном случае это «Московский Областной ОМС»,

«Территориальный Фонд ОМС» и «ОТР». Контроль качества выполняемой работы определяются внутренними правилами, действующими в «СофтТраст».

Осуществление деятельности на диаграммах отображаются, как нижние стрелки. На контекстной диаграмме «Обработка НСИ» они представлены статистами медицинских организаций, что означает полное отсутствие автоматизированной обработки пакетов НСИ.

Любой процесс должен иметь входные данные, на диаграммах такая информация представлена стрелками, которые входят в блок слева и олицетворяют все то, что непосредственно обрабатывается в процессе работы.

В данной диаграмме входящие данные представлены в виде пакетов НСИ разных форматов и данных, необходимых для дальнейшей обработки имеющихся пакетов НСИ.

Выходящие данные представлены на контекстной диаграмме в загруженных в базу данных актуальных пакетах НСИ. Блок «Обработка НСИ» представлен на рисунке 3.

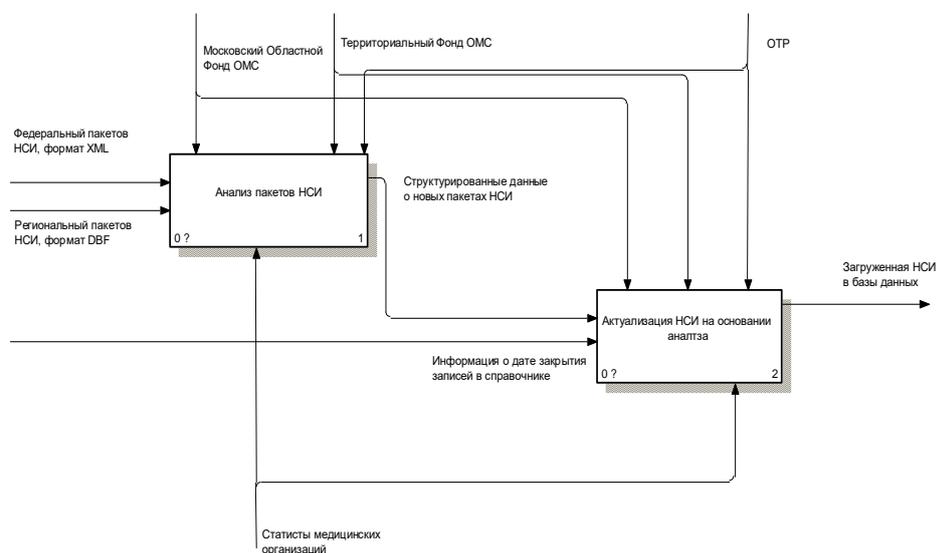


Рисунок 3 - Декомпозиция контекстной диаграммы «Обработка НСИ»

Блок «Обработка НСИ» декомпозируется на «Анализ пакетов НСИ», «Актуализация НСИ на основе анализа».



Поступающая информация – структурированные данные о новых пакетах НСИ, информация на выходе – загруженная в базу данных НСИ. Данная модель представлена на рисунке 5.

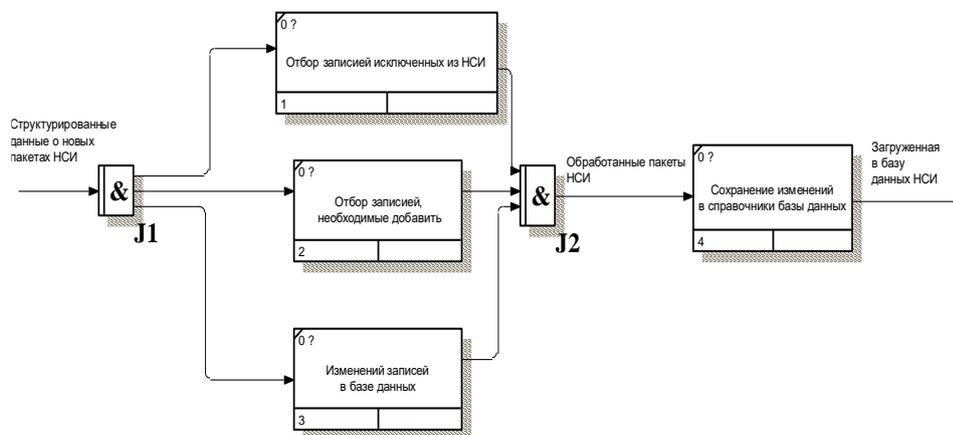


Рисунок 5 - Декомпозиция блока «Актуализация НСИ»

В результате, в каждой больнице изменение НСИ происходит локально, сразу на рабочем месте. Данная модель распространения и обновления НСИ не является эффективной из-за различной скорости обработки ЛПУ (лечебно-профилактическое учреждение) НСИ, а также наличие человеческого фактора. Схема распространения НСИ представлена на рисунке 6.

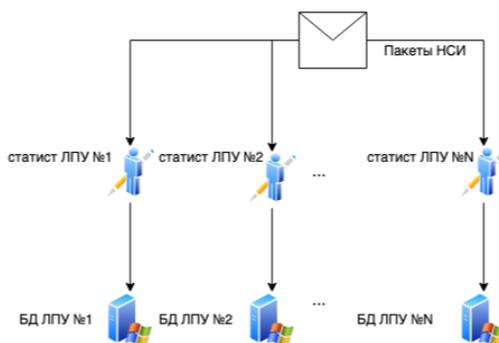


Рисунок 6 – Схема распространения НСИ

Данная модель распространения и обновления НСИ не является эффективной из-за различной скорости обработки ЛПУ (лечебно-

профилактическое учреждение) НСИ, а также из-за наличия человеческого фактора.

Процесс информационного обмена (выгрузка реестров счетов) необходим для осуществления функции ТФОМС МО по автоматизированной обработке полученных от медицинских организаций сведений о медицинской помощи, оказанной застрахованным лицам.

Процесс информационного обмена состоит из последовательно выполняемых этапов:

Этап 0 (вспомогательный). Определение страховой принадлежности застрахованного лица при обращении в медицинскую организацию.

Этап 1. Передача и получение сводного отчета об оказанной застрахованным лицам медицинской помощи.

Этап 2. Форматно-логический контроль целостности данных объектов информационного обмена.

Этап 3. Идентификация застрахованного лица по региональному сегменту единого регистра застрахованных лиц, определение страховой медицинской организации, ответственной за оплату счета;

Этап 4. Выявление застрахованных лиц, которым оказана медицинская помощь вне территории страхования, определение их территории страхования и реквизитов действующего полиса ОМС.

Для информационного обмена между МО и ТФОМС МО используется «Пакет информационного обмена» (далее - Пакет). При информационном обмене каждый Пакет нумеруется, и весь информационный обмен в рамках Пакета происходит с указанием этого номера. Последовательная нумерация Пакетов ведется в течение даты запроса.

Все пакеты, полученные ТФОМС МО от МО, обрабатываются в порядке очереди поступления – до успешного завершения форматно-логического контроля предыдущего пакета, следующий пакет не обрабатывается. Пакет является логически-неделимой единицей приёма данных от МО. Он может быть принят или не принят только целиком. В Пакет должен быть включен

единственный электронный документ - Свод реестров обратившихся пациентов (далее - Свод). Каждый Свод является ZIP-архивом, содержащим файлы реестров с персональными данными застрахованных лиц, обратившихся в любую МО, входящую в юридическое лицо [2].

Предусмотрено два режима выполнения этапа информационного обмена – проверочный (тестовый) и контрольный.

Информационный обмен в проверочном режиме осуществляется с целью заблаговременного контроля, сверки и идентификации персональных данных застрахованных лиц по мере персонифицированного учета медицинской помощи, оказанной им, и обеспечивает своевременную идентификацию большинства застрахованных лиц с учетом сроков обработки запросов на определение страховой принадлежности в ЦС ЕРЗ и сроков выставления счетов на оплату в СМО. Проверочный режим доступен начиная с 10-го числа месяца отчетного периода.

Проверочный режим делается недоступным после поступления за отчетный период пакета в контрольном режиме при условии безошибочной обработки такого пакета. Проверочный режим снова становится доступным после санкционирования пересдачи. С первого рабочего дня месяца, следующего за отчетным, проверочный режим осуществляется только по Этапам 1 и 2 [16].

Информационный обмен в контрольном режиме производится с целью окончательного приемо-сдаточного контроля, сверки и идентификации персональных данных застрахованных лиц.

Информационный обмен в контрольном режиме производится ежемесячно, начиная с первого рабочего дня месяца, следующего за отчетным.

При отсутствии попыток медицинской организации сдать Пакет электронного отчета в ТФОМС МО в контрольном режиме до пятого рабочего дня месяца, следующего за отчетным, ТФОМС МО оставляет за собой право считать, что в данной медицинской организации медицинская помощь лицам, застрахованным по обязательному медицинскому страхованию, в отчетном периоде не оказывалась. Прикладному программному обеспечению (работу

приемки счетов) ТФОМС МО будет дана команда о прекращении ожидания отчета от конкретной «неактивной» медицинской организации. Если медицинская организация, включенная в список «неактивных», попытается впоследствии отправить в ТФОМС МО Пакет электронного отчета в контрольном режиме, то в приёме Пакета ей будет отказано с разъяснением о вышеизложенной причине.

В случае выгрузки реестров счетов, контекстная диаграмма «как есть» предоставлена на рисунке 7.

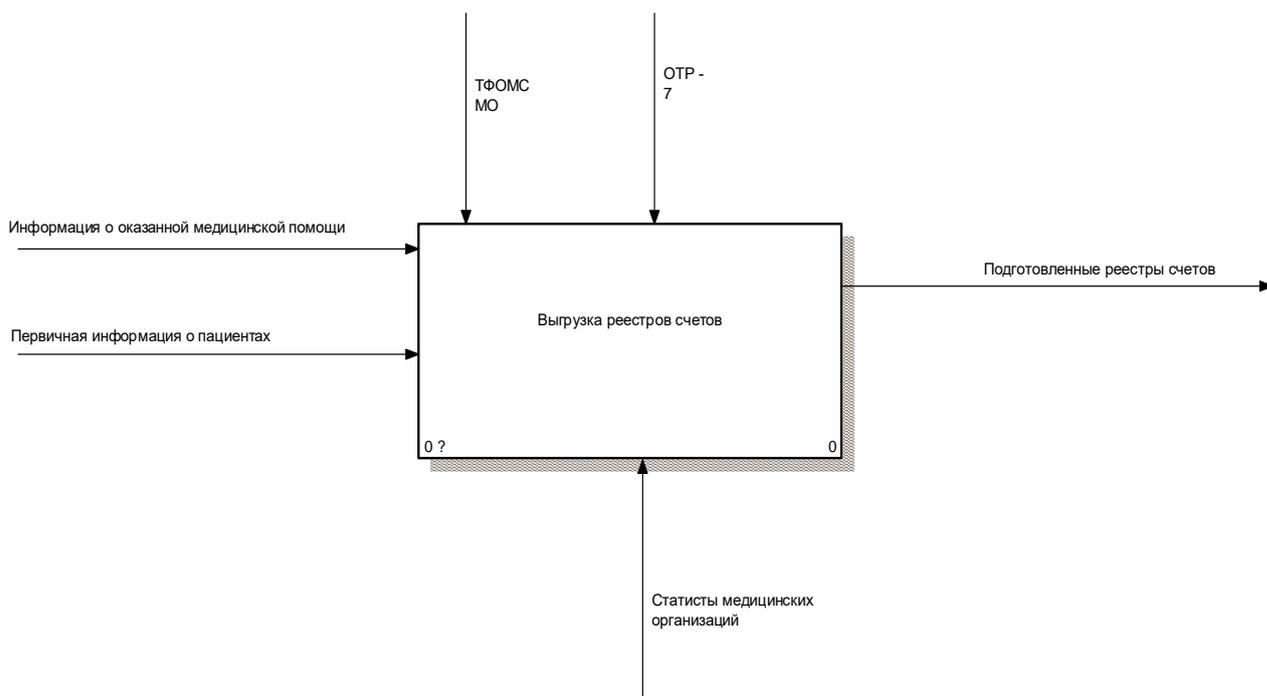


Рисунок 7 – Контекстная диаграмма «Выгрузки реестров счетов»

Входными параметрами являются информация об оказанной медицинской помощи и первичная информация о пациенте (СНИЛС, тип места жительства, социальный статус и другие). Управляющие процессы ОТР – 7, важно опираться только на последнюю версию ОТР для поддержания актуальности процессов контролей, ТФОМС МО. ОТР устанавливает требования к регламенту информационного обмена в соответствии с «Порядком ведения персонафицированного учета в сфере обязательного медицинского страхования», утвержденным Приказом Минздравсоцразвития России от 25.01.2011 №29н, «Техническими условиями информационного взаимодействия

автоматизированных информационных систем участников ОМС в Московской области», утвержденными Приказом ТФОМС МО от 17 ноября 2014 года № 296 , Соглашение между Министерством здравоохранения Московской области и Территориальным фондом обязательного медицинского страхования Московской области о предоставлении межбюджетных трансфертов из бюджета Московской области бюджету Территориального фонда обязательного медицинского страхования Московской области на дополнительное финансовое обеспечение организации обязательного медицинского страхования и мероприятий в рамках Московской областной программы государственных гарантий бесплатного оказания гражданам медицинской помощи.

Результатом являются подготовленные реестры счетов. Механизм осуществления деятельности – статисты медицинских организаций.

Декомпозиция контекстной диаграммы «Выгрузка реестров счетов» представлена на рисунке 8.

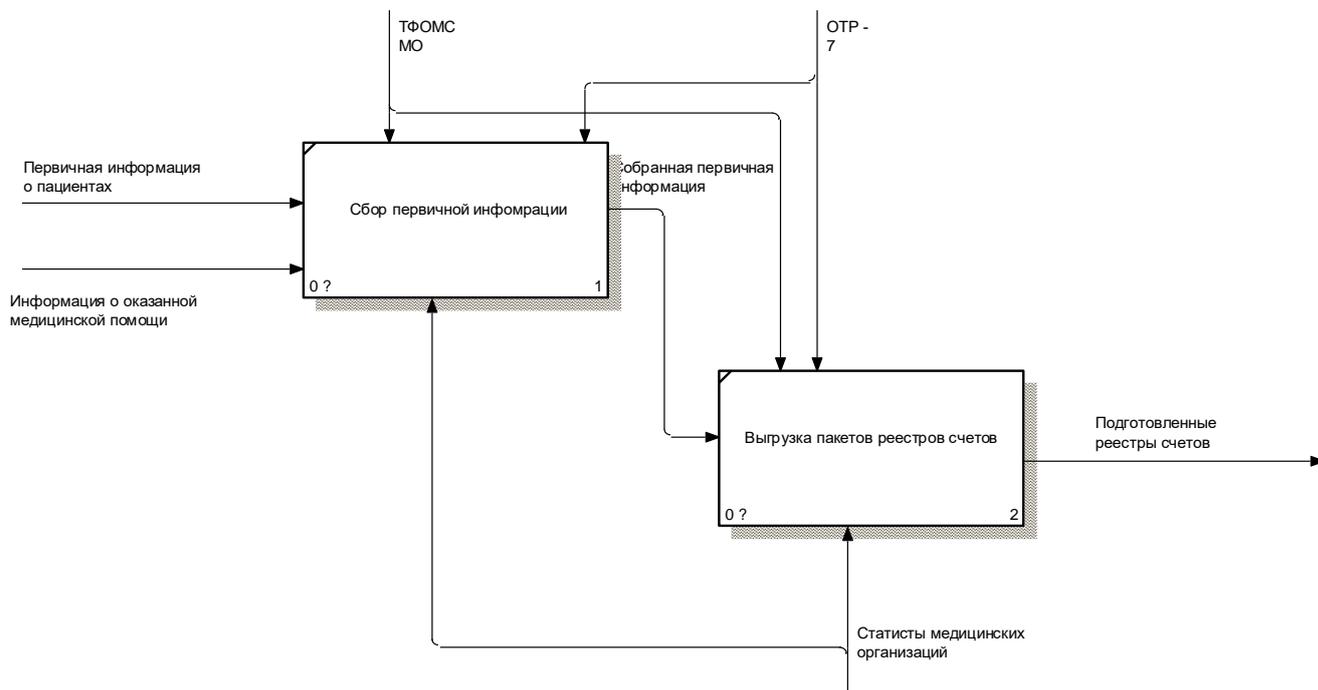


Рисунок 8 – Декомпозиция блока «Выгрузка реестров счетов»

Первым действием собирается вся информация о оказанной медицинской помощи с учетом найденных первичны данных пациента. Данные в соответствии регламента (ОТР – 7) формируются в таблицы dbf и выгружаются в папку

«Reestr» (расположена в корне папки ТМ:МИС) для дальнейшей отправки на сдачу статистами медицинских организаций.

Статистам медицинских организаций в следствии больших объемов данных пациентов, необходимо потратить много времени для проверки на корректность оформления медицинского случая, потраченное время, чаще всего превышает отведенное для сдачи реестров счетов, в результате чего, больница получает снятия по некорректно оформленным случаям.

Исходя из выше представленных диаграмм, можно сделать вывод, о необходимости создать автоматизированный протокол обработки НСИ и выгрузки реестров счетов в медицинских организациях.

### 1.3 Постановка задачи

Целью создания протоколов является усовершенствование деятельности отдела взаиморасчетов, при выгрузке реестров счетов, за счет поддержания автоматизированной обработки пакета НСИ, что даст возможность автоматического изменения НСИ с затратой минимального количества времени, уменьшить количество ошибок в показателях реестров во время контрольной сдачи счетов, возможность повысить защищенности данных (заниматься обработкой будет теперь один человек), а также усовершенствовать процесс автоматизации получения первичной информации.

Основным пользователем разрабатываемых протоколов являются статисты в медицинских организациях.

Разработка протоколов состоит из следующих, ниже перечисленных, задач.

Предпроектная подготовка – первым этапом является изучение предметной области, уточнение целей и задач. Изучение работы компании, анализ ОТР.

Разработка технического задания – вторым этапом разработки является утверждение окончательного технического задания, с учетом всех требований,

что касается согласования правил контроля и требований к технической части проекта.

Разработка протоколов – третий этап включает в себя разработку протоколов загрузки НСИ с учетом особенностей ТФОМС МО и МФОМС, а также протокол выгрузки реестров счетов с учетом анализа ОТР.

Результатом протокола выгрузки реестров счетов должен являться свод реестров обратившихся пациентов.

В Свод реестров обратившихся пациентов должны входить файлы информационного обмена с уникальным в рамках Свода именем. Имя файлу информационного обмена может быть присвоено произвольным образом с учетом ограничений файловой системы Microsoft Windows. Длина имени файла, включая расширение имени и разделитель – точку между именем и расширением, не должна превышать 50 символов. Количество файлов, включенных в Свод, ограничено только максимальным размером Свода - 20 Мб.

В реестр счетов должны включаться следующие электронные документы.

- таблица пролеченных пациентов в формате PCCMMYY.DBF;
- таблица медицинских услуг в формате UCCMMYY.DBF;
- таблица исполнителей медицинских услуг в формате DCCMMYY.DBF;
- таблица учета времени оказания скорой медицинской помощи в формате XCCMMYY.DBF;
- таблица медицинские мероприятия первого этапа диспансеризации взрослого населения в формате MCCMMYY.DBF;
- таблица учета специализированной медицинской помощи в стационарных условиях и в условиях дневного стационара на основе клинико-статистических групп (КСГ) в формате GCCMMYY.DBF;
- таблица формула зуба в формате ZCCMMYY.DBF;
- таблица сведения о диспансеризации, профилактических осмотрах и назначениях в формате NCCMMYY.DBF;

– таблица сведения о лечении онкологического заболевания или медицинской помощи при подозрении на злокачественное новообразование в формате CCCMMYY.DBF.

В случае, если медицинская организация оказывает и представляет к оплате медицинские услуги скорой медицинской помощи, а также все остальные необходимые данные, в Свод по плательщику за отчётный период должен быть включен непустой (содержащий записи) файл информационного обмена [16].

Протокол загрузки НСИ должен обрабатывать файлы МОФОМС МО из таблицы 1.

Таблица 1 – Федеральные справочники

Наименование справочника/классификатора	Имя файла
Классификатор профилей оказанной медицинской помощи	V002MMYY.XML
Классификатор медицинских специальностей	V021MMYY.XML
Классификатор видов медицинской помощи	V008MMYY.XML
Классификатор типов документов, подтверждающих факт страхования по ОМС	F008MMYY.XML
Классификатор типов документов, удостоверяющих личность	F011MMYY.XML
Классификатор условий оказания медицинской помощи	V006MMYY.XML
Классификатор результатов обращения за медицинской помощью	V009MMYY.XML
Классификатор исходов заболевания	V012MMYY.XML
Классификатор способов оплаты медицинской помощи	V010MMYY.XML
Классификатор профиля койки	V020MMYY.XML
Классификатор дополнительных классификационных критериев	V024MMYY.XML
Классификатор клинико-статистических групп	V023MMYY.XML
Классификатор целей посещения (КРС)	V025MMYY.XML
Классификатор клинико-профильных групп	V026MMYY.XML

Продолжение таблицы 1 – Федеральные справочники

Классификатор противопоказаний и отказов	N001MMYY.XML
Классификатор стадий	N002MMYY.XML
Классификатор Tumor	N003MMYY.XML
Классификатор Nodus	N004MMYY.XML
Классификатор Metastasis	N005MMYY.XML
Справочник соответствия стадий TNM	N006MMYY.XML
Классификатор гистологических признаков	N007MMYY.XML
Классификатор соответствия гистологических признаков диагнозам	N009MMYY.XML
Классификатор маркёров	N010MMYY.XML
Классификатор значений маркёров	N011MMYY.XML
Классификатор соответствия маркёров диагнозам	N012MMYY.XML
Классификатор типов лечения	N013MMYY.XML
Классификатор типов хирургического лечения	N014MMYY.XML
Классификатор линий лекарственной терапии	N015MMYY.XML
Классификатор циклов лекарственной терапии	N016MMYY.XML
Классификатор типов лучевой терапии	N017MMYY.XML
Классификатор поводов обращения	N018MMYY.XML
Классификатор целей консилиума	N019MMYY.XML
Классификатор лекарственных препаратов, применяемых при проведении лекарственной терапии	N020MMYY.XML
Классификатор соответствия лекарственного препарата схеме лекарственной терапии	N021MMYY.XML
Справочник «Классификатор НРУЗ ФОМС»	V001MMYY.XML
Классификатор видов направления	V028MMYY.XML
Классификатор методов диагностического исследования	V029MMYY.XML
Единый реестр страховых медицинских организаций, осуществляющих деятельность в сфере обязательного медицинского страхования. Реестр СМО	F002MMYY.XML
Общероссийский классификатор административно-территориального деления ( ОКАТО )	O002MMYY.XML

Для справочников ТФОМС МО список загружаемых файлов представлен на таблице 2.

Таблица 2 – Справочники ФТОМС МО

Наименование справочника/классификатора	Имя файла
Медицинские организации	LPUMMY.Y.DBF
Медицинские услуги	MDUMMY.Y.DBF
Международный классификатор болезней (МКБ10)	MKBMMYY.DBF
Тарификатор медицинских услуг	TMUMMY.Y.DBF
Квартальное (и месячное) распределение объемов медицинской помощи	QMVMYY.Y.DBF
Базовая ставка и коэффициенты для оплаты по КСГ	KDGMYY.Y.DBF
Сборник элементов кодированных перечней	SBRMMYY.DBF
Справочник взаимного соответствия исходов заболевания и результатов обращения за медицинской помощью	FLK1MMYY.DBF
Формула зубов по ВОЗ	ZFVMYY.Y.DBF

Интеграция разработанных протоколов в медицинские организации - четвертый этап, в который входят процедуры распространения протоколов по медицинским организациям и назначения прав доступа.

В данном разделе были описаны характеристика предметной области, структура предприятия, была обоснована необходимость создания протоколов обработки НСИ и выгрузки реестров счетов, разработаны диаграммы «КАК ЕСТЬ», а также поставлены задачи, необходимые выполнить при разработке протоколов.

## 2 Обоснование проектных решений

### 2.1 Обоснование проектных решений по техническому обеспечению

Техническое обеспечение – представляет собой совокупность технических средств, компьютерной техники, средств передачи информации, используемых в автоматизированных системах управления и в информационных системах.

Далее следует привести структуру технического обеспечения.

В состав базового технического обеспечения входят такие компоненты как:

- микропроцессор;
- постоянно запоминающее устройство;
- оперативная память – ОЗУ;
- регистровая память;
- видеопамять;
- блок питания.

Периферийное техническое обеспечение предназначено для решения прикладных задач пользователя. Состоящее из таких компонентов как:

- устройства ввода (клавиатура, мышь, сканер, джойстик и др.);
- устройства вывода (дисплей, принтер и др.);
- устройства (накопители) внешней памяти (дискета, CD и др.).

Специализированное техническое обеспечение состоящие из устройств, используемые для решения уникальных проблем.

Для функционирования отдела информационных технологий в рамках протоколов выгрузки реестров счетов и загрузки НСИ потребуются следующие элементы технического обеспечения:

Программного комплекса необходимо следующее программное обеспечение, для рабочей станции:

- Операционная система MS Windows XP SP3, MS Win Vista, Windows 7, Windows 8;

- MS .NET Framework 4.0;
- MS Office 2003, 2007, 2010, 2013;
- MS Internet Explorer 9.0 или выше.

Для тонкого клиента:

– Операционная система MS Windows XP SP3, MS Win Vista, Windows 7, Windows 8, Unix;

- подобные операционные системы.

Для сервера:

- Процессор Intel-совместимый;
- 2048 Mb RAM;
- Объем свободного дискового пространства – 40 Gb;
- Наличие сетевой карты;
- Наличие USB-порта.

Для рабочей станции:

- Процессор Intel-совместимый;
- 1024 Mb RAM;
- Объем свободного дискового пространства – 4 Gb;
- Наличие сетевой карты;
- Наличие USB-порта;
- Принтер лазерный/струйный формата А4.

## 2.2 Обоснование проектных решений по информационному обеспечению

Информационное обеспечение является средством для решения таких задач как:

- осуществление возможности на основе кодирования объектов однозначного и экономичного представления информации;
- на основе классификации объектов реализовывать процедуры анализа и обработки информации с учетом характера связей между объектами;

– на основе экранных форм ввода-вывода данных организовывать взаимодействия пользователей с системой;

– на основе унифицированной системы документации обеспечить эффективного использования информации в контуре управления деятельностью объекта автоматизации.

Информационное обеспечение состоит из компонентов, включающих в себя: внешнее информационное обеспечение, которое подразумевает под собой классификаторы технико-экономической информации, документы, методические инструктивные материалы. И внутреннее информационное обеспечение, которое подразумевает под собой макеты/экранные формы для ввода первичных данных в ЭВМ или вывода результатной информации, структуры информационной базы: входных, выходных файлов, базы данных.

Требования, предъявляемые к информационному обеспечению:

– возможность поддержания всех автоматизируемых функций объекта;

– кодирования информации должно осуществляться принятыми у заказчика классификаторами;

– обеспечение совместимости с информационным обеспечением систем, взаимодействующих с разрабатываемой системой;

– формы документов должны отвечать требованиям корпоративных стандартов заказчика;

– структура документов должны соответствовать характеристикам терминалов на рабочих местах конечных пользователей;

Информационное обеспечение (ИО) включает в себя: систему классификации и кодирования, систему унифицированной документации, используемой в ИО и информационную базу.

Классификатор - это структурированный подбор наименований группировок, состоящих из объектов, признаков и их кодовых обозначений. Классификаторы используют как инструмент описания данных, кодирования информации и предназначены для обеспечения машинной обработки и выдачи данных в удобной форме. В зависимости от применения они делятся на три

группы: общегосударственные классификаторы, отраслевые (ведомственные) классификаторы, используемые в пределах определенной отрасли (ведомства) и локальные классификаторы, используемые в пределах организации или группы организации.

Для разработки протоколов выгрузки реестров счетов и загрузки НСИ будет использоваться только локальный классификатор. Значительную долю немашинного информационного обеспечения составляет документация. В условиях автоматизации важное значение придается унификации документации, устанавливающей единые требования к содержанию и построению документов. Унификация представляет собой отбор необходимой номенклатуры документов.

Существует три способа организации информационной базы (ИБ): файловая; интегрированная и смешанная организация.

Под файловой организацией ИБ понимается локальное размещение базы на компьютере, доступ к которому других пользователей осуществляется стандартными методами ОС для обмена данными по сети. Под смешанной организацией ИБ подразумевается распределённая база данных, хранящаяся на нескольких серверах и реплицирующая изменения в каждой из них по расписанию, данная структура ИБ используется в системах класса ERP для работы в одной ИБ территориально удалённым офисам одновременно.

Интегрированный способ организации ИБ представляет собой совокупность взаимосвязанных и хранящихся вместе данных при такой минимальной избыточности, которая допускает их использование оптимальным образом для любых приложений и при этом обеспечивается независимость данных от программы, а для актуализации данных используется общий способ управления.

Учитывая все вышеприведенные особенности характеристик, наиболее целесообразной организацией ИБ является интегрированная, так как размер базы будет постепенно увеличиваться. И оптимальным выбором будет использование СУБД вместо файлового хранения базы данных

Существует три модели логической структуры базы данных: иерархическая, сетевая и реляционная.

При использовании иерархической модели каждой информационной единице (сегменту), кроме корневого, соответствует один исходный сегмент и между исходным и порожденным сегментом устанавливается только одна связь. Такая структура удобна для отображения отношений типа «один ко многим» в предметной области. Просмотр иерархической структуры возможен только с корневой вершины. Пропуск сегмента в иерархическом пути при доступе к заданному сегменту не допускается. Основные недостатки иерархической структуры: трудность (неэффективность) отображения отношений типа «многие ко многим»; длительность доступа к сегментам, находящимся на нижних уровнях иерархии; ориентированность на определенный тип (разрез) запроса.

Реляционная база данных - это множество отношений. Реляционная модель основана на математической логике и является простейшей и наиболее привычной формой представления данных в виде таблицы. Строка таблицы эквивалентна записи файла базы данных, а колонка - полю записи. Доступ к элементу данных осуществляется посредством связи требуемой строки (записи) с требуемой колонкой (полем). Достоинством реляционной модели является сравнительная простота инструментальных средств ее поддержки, недостатком - жесткость структуры данных (например, невозможность задания строк таблицы произвольной длины) и зависимость скорости ее работы от размера базы данных.

Преимущества использования реляционных базы данных состоит в следующем:

Простота - в реляционной модели данных существует всего одна информационная конструкция, которая формализует табличное представление данных, привычное для пользователей;

Теоретическое обоснование - наличие теоретически обоснованных методов нормализации отношений позволяет получать базы данных с заранее заданными свойствами (в основном, с гарантией минимальной избыточности представления данных);

Независимость данных - когда необходимо изменить структуру реляционной базы данных, то это приводит к минимальным изменениям в программном продукте.

Моделью логической структуры базы данных является именно реляционная, так как она позволяет довольно быстро сформировать связи между таблицами для правильного построения запросов к базе данных и также легко разорвать эти связи и создать новые для построения другого запроса. Кроме того, архитектура построения связи более проста и время выполнения запроса в реляционной модели выше чем при использовании сетевой или иерархической структуры.

### 2.3 Обоснование проектных решений по программному обеспечению

Программное обеспечение – основная часть компьютерной системы. Что подразумевает под собой продолжение технических средств. ПО современных компьютеров состоит из множества программ – от игровых до научных.

Обоснование проектных решений по программному обеспечению заключается в формировании требований к системному и специальному (прикладному) программному обеспечению и выборе на основе этих требований соответствующих компонентов программного обеспечения.

Программное обеспечение делится на 2 типа: системное и прикладное. Системное программное обеспечение – включает в себя набор программ, которые управляют такими компонентами компьютера, как процессор, коммуникационные и периферийные устройства. К прикладному программному обеспечению относятся программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы.

В свою очередь системное программное обеспечение разделяют на базовое ПО и сервисное ПО. Базовое программное обеспечение состоит из:

– операционной системы, которая предназначена для управления выполнением пользовательских программ, планирования и управления вычислительными ресурсами ЭВМ;

– операционной оболочки, которая состоит из специальных программ, предназначенные для облегчения общения пользователя с командами операционной системы. Операционные оболочки имеют текстовый и графический варианты интерфейса конечного пользователя;

– сетевой операционной системы, состоящей из комплекса программ, обеспечивающий обработку, передачу и хранение данных в сети. Данная операционная система предоставляет пользователям различные виды сетевых служб и поддерживает работу в абонентских системах.

В свою очередь прикладное программное обеспечение дает возможность обеспечить применение вычислительной техники в различных сферах деятельности. Помимо создания новых программных продуктов, разработчики прикладных программ большие усилия тратят на совершенствование и модернизацию популярных систем, создание их новых версий.

Для отображения полноты представления, структура прикладного программного обеспечения состоит из программных средств общего назначения, программных средств специального назначения и программных средств профессионального уровня.

Далее следует подробно описать используемое программное обеспечение компании ООО «СофтТраст». В состав ПО данного предприятия входит широкий перечень программных продуктов, которые позволяют полноценно функционировать и решать любые задачи [17].

Цель создания протоколов загрузки НСИ и выгрузки реестров счетов заключается в усовершенствовании работы отдела взаиморасчетов, что включает в себя возможность вносить и изменять информацию о НСИ и корректной выгрузки реестров счетов с затратой минимального количества времени и усилий. Проведя анализ имеющегося на предприятии оборудования и

программного обеспечения, требуется разработать современный программный продукт, для которого были выбраны следующие программные средства:

Среда SQL Server Management Studio. Это единая универсальная среда для доступа, настройки и администрирования всех компонентов MS SQL Server, а также для разработки компонентов системы, редактирования текстов запросов, создания скриптов и пр. Благодаря наличию большого количества визуальных средств управления, «Среда SQL Server Management Studio» позволяет выполнять множество типовых операций по администрированию MS SQL Server администраторам с любым уровнем знаний SQL Server. Удобная среда разработки, встроенный веб-браузер для быстрого обращения к библиотеке MSDN или получения справки в сети, подробный учебник, облегчающий освоение многих новых возможностей, встроенная справка от сообществ в Интернете и многое другое позволяют максимально облегчить процесс разработки в среде SQL Server, а также дает богатые возможности для создания различных сценариев SQL Server.

Сервер баз данных Microsoft SQL Server в качестве языка запросов использует версию языка SQL, получившую название Transact-SQL (сокращённо T-SQL). T-SQL позволяет использовать дополнительный синтаксис для хранимых процедур и обеспечивает поддержку транзакций (взаимодействие базы данных с управляющим приложением).

В качестве клиентской части выступает Медицинская информационная система «ТМ:МИС» (ТрастМед:Медицинская информационная система) предназначенная для комплексной автоматизации деятельности персонала лечебно-профилактических учреждений. Для функционирования

МИС, как совокупность программного, технического и организационно методического обеспечения, позволяет сохранять информацию в блоках памяти и внешних запоминающих устройствах во всех режимах, кроме режима записи данных, в соответствии с эксплуатационной документацией на внешние запоминающие устройства, при включениях и отключениях, в том числе аварийных, технических средств. Аварийный режим функционирования

системы возникает в случае возникновения аварийной ситуации. К аварийным ситуациям относятся:

- отказ системы;
- сбой системы.

Отказом системы следует считать событие, состоящее в утрате работоспособности системы и приводящее к невыполнению или неправильному выполнению тестов, лицензионного программного обеспечения или задач функциональных подсистем. Сбоем системы следует считать событие, состоящее во временной утрате работоспособности системы и характеризующееся возникновением ошибки при выполнении тестов, лицензионного программного обеспечения или задач функциональных подсистем. В случае перехода системы в предаварийный режим необходимо оповестить пользователей о завершении работы всех приложений в минимально возможный срок, с сохранением данных. Затем должны быть произведены следующие работы:

- выключение рабочих станций операторов;
- выключение всех периферийных устройств;
- резервное копирование БД.

После этого выполняется комплекс мероприятий по устранению причины/причин перехода МИС в аварийный режим. Время для восстановления работоспособности не должно превышать 48 часов.

Средства диагностирования МИС обеспечивают выявление неработоспособности программных средств системы, а именно:

- отчетов;
- протоколов импорта/экспорта данных;
- ссылочной целостности базы данных.

Система в целом сохраняет работоспособность при некорректных действиях пользователя и обеспечивает полную защищенность логической и физической целостности данных. Имеет встроенные механизмы диагностики целостности данных и корректной работы подсистем. Система обладает надежностью, обеспечивающей работу пользователей в круглосуточном режиме

и оперативное восстановление работоспособности при сбоях. Надежность МИС обеспечивается комплексом мероприятий отладки, поиска и исключения ошибок, а также процедурами резервирования информационных массивов. Для обеспечения надежности МИС предусмотрены функции формирования резервных копий баз данных, программных процедур и возможности оперативного восстановления резервных данных и программных процедур.

МИС должна функционировать как в автономном режиме рабочей станции, так и в составе локальной сети (с использованием одной из рабочих станций в качестве сервера БД).

#### 2.4 Обоснование проектных решений по технологическому обеспечению

Технологическое обеспечение представляет собой совокупность методов и средств сбора, обработки, хранения, передачи и защиты информации на базе вычислительной техники. Технологическое обеспечение в свою очередь состоит из подсистем, которые обеспечивает технологию функционирования. Сюда входит:

- информационное состоит из множества управленческих документов и их компьютерных носителей, которые применимы для автоматизации управления, что дает возможность решать вопросы, касающиеся решений по объемам, размещению, формам организации информации, циркулирующей в ИТ и поступающей из внешней среды;

- техническое состоит из комплекса технических средств, таких как сбор, регистрация, хранение, поиск, защита, передача, обработка, отображения и размножения информации и обеспечивает работу;

- математическое состоит из математических методов и алгоритмов обработки информации, которые используются в системе при решении функциональных задач;

- программное представляет собой совокупность программ, состоящую из системного и прикладного программного обеспечения, которые в свою очередь

реализуют функции и задачи, и обеспечивающих устойчивую работу комплексов технических средств;

– лингвистическое объединяет в себе совокупность традиционных языков, научно-технических терминов и языковых средств формализации естественного языка;

– организационное состоит из совокупности средств и методов организации производства и управления им в условиях внедрения и эксплуатации ИТ;

– правовое представляет собой совокупность правовых норм;

– эргономическое подразумевает под собой совокупность методов и средств, которые используются на разных этапах разработки и функционирования;

– кадровое характеризуется совокупностью методов и средств по организации и проведению обучения персонала и приемам работы с ИС.

Технологический процесс обработки данных – представляет собой комплекс операций, направленных на преобразование информации в соответствии с поставленной целью с момента входа в информационную систему до момента ее потребления пользователями. Чтобы понять и определить сложность и многообразие вариантов технологических процессов следует разделить на этапы и операции.

Операция сбора информации заключается в обеспечении системы управления необходимым объемом данных, что позволит выполнить нужные задачи. В ходе работы над сбором информации производятся измерение, подсчет, взвешивание материальных объектов, подсчет денежных купюр, получение временных и количественных характеристик работы отдельных исполнителей.

Операция передача информации заключается в обмене данными и переносе информации в пространстве. Данную операцию можно осуществить несколькими способами: с помощью курьера, пересылки по почте, доставки транспортными средствами, дистанционной передачи по каналам связи, с

использованием других средств коммуникаций. Преимущество дистанционного вида передачи данных по каналам связи заключается в сокращении времени их движения. Наиболее удобным и выгодным находят использование технических средств сбора и регистрации, которые в свою очередь, автоматически собирая информацию с установленных на рабочих местах датчиков, осуществляют передачу в ЭВМ для последующей обработки, что повышает ее достоверность и снижает трудоемкость.

Операция машинного кодирования заключается в машинном представлении информации на машинных носителях с помощью кодов, принятых в компьютере. Кодирование данных осуществляется путем переноса информации первичных документов на магнитные диски, в дальнейшем информация вводится в компьютер для обработки. Запись информации на машинные носители осуществляется на компьютере как самостоятельная процедура или как результат обработки.

Так как, «СофтТраст» ранее не имел протоколов загрузки НСИ и выгрузки реестров счетов, продуктивность работы отдела взаиморасчетов, напрямую зависела от реализации данных процессов статистами медицинских организация. Необходимо провести подробный анализ и выявить основные достоинства и недостатки ручного обновления НСИ и выгрузки реестров счетов.

К преимуществам ручного обновления НСИ можно отнести такие свойства как:

- актуализация пакетов НСИ;
- личный анализ изменений в НСИ;
- отсутствие зависимости от обновлений или реструктуризации ЦОДа.
- К преимуществам ручной выгрузки реестров счетов можно отнести:
  - формирования случая оказания медицинской помощи специализированными врачами;
  - отсутствие зависимости от обновлений или реструктуризации ЦОДа.

Далее следует обозначить недостатки ручного обновления НСИ и выгрузки реестров счетов, к ним относят:

– отсутствие централизованного обновления в базах данных различных медицинских организаций;

– человеческий фактор (допущение ошибок в обработке НСИ и проверки контролей согласно ОТР-7);

– для корректной и долгосрочной работы обновления НСИ, данный процесс должен осуществляться исключительно статистами медицинских организаций, обычно это один человек на медицинскую организацию;

– необходимость в поддержке обновления НСИ. Этот фактор обусловлен тем, что в ходе обновления или анализа НСИ существует вероятность возникновения вопросов и ошибок, исходя из этого вытекает потребность в оперативности их решения. Такими проблемами как правило являются: сбой конфигурации, потребность в консультации персонала относительно работы с программой и установка обновлений программы. Способ решения проблемы - обратится в техническую поддержку с интересующим вопросом. Ответ на поставленный вопрос пользователь получит в течении 5-ти рабочих дней.

– большой расход времени на обновление НСИ и выгрузку реестров счетов (до 10 рабочих дней).

Учитывая все вышеприведенные факторы и исходя из того, что недостатков выявлено в разы больше, для совершенствования работы отдела взаиморасчетов следует принять решение о разработке протоколов выгрузки реестров счетов и загрузки НСИ, которое позволит автоматически анализировать все пакеты НСИ, изменять информацию в БД, составлять отчет анализа изменений НСИ, производить массовую проверку на контроли согласно ОТР-7, ускорить процесс обработки НСИ и выгрузки реестров счетов, за счет комплексной обработки данных.

Для наглядности необходимости в разработке протоколов выгрузки реестров счетов и загрузки НСИ, с целью совершенствования работы сотрудников отдела взаиморасчетов, следует выявить преимущества и слабые стороны данных протоколов. К достоинствам относят такие факторы как:

- отсутствие в необходимости локального изменений данных в НСИ. Для эффективной и полноценной работы требуется доступ к серверу;
- нет потребности в специальной настройке и администрировании, администраторами являются разработчики;
- автоматическое обновление версий протоколов;
- с учетом того, что конфиденциальные данные хранятся на удаленных серверах, существуют средства надежной защиты данных клиентов.

Одним из недостатков протоколов является трудность в анализе и правильной трактовке медицинской терминологии, прописанной в ОТР.

## 3 Программная реализация проектных решений

### 3.1 Разработка функциональной модели

При проектировании протоколов обработки нормативно-справочной информации и выгрузки реестров счетов для медицинской организации МО было выбрано инструмент AllFusion Process Modeler (далее BPwin) — CASE-средство для реализации процессов документирования, моделирования, анализа и др.

BPwin имеет возможность создавать диаграммы в нотации IDEF0, IDEF3, DFD, в процессе реализации, данный инструмент позволяет переключаться между нотациями, например, с нотации IDEF0 в любом разделе модели на нотацию IDEF3 или DFD, а также предоставляет возможность создания смешанной модели.

Первые шаги работы с BPwin начинается с создания новой диаграммы, для которой нужно указать имя и тип. От выбранного типа модели будет зависеть в каких нотациях можно производить декомпозицию работ.

К примеру, если выбрать тип «Business Process» (IDEF0), то в созданной модели можно производить декомпозицию работ в нотациях IDEF0, IDEF3 или же DFD; если выбран тип «Data Flow» (DFD) — в нотациях DFD и IDEF3; если же выбран тип «Process Flow» (IDEF3) — то только в нотации IDEF3.

Первым этапом была создана контекстная диаграмма, которая описывает систему в целом, входные данные и ее взаимодействие с другими ресурсами. После чего контекстная диаграмма была декомпозирована для подробного представления внутренних механизмов.

Данный процесс продолжается до тех пор, пока не была достигнута желаемая степень детализации.

На рисунке 9 представлена контекстная диаграмма бизнес-процесса протокола обработки НСИ для медицинской организации МО.

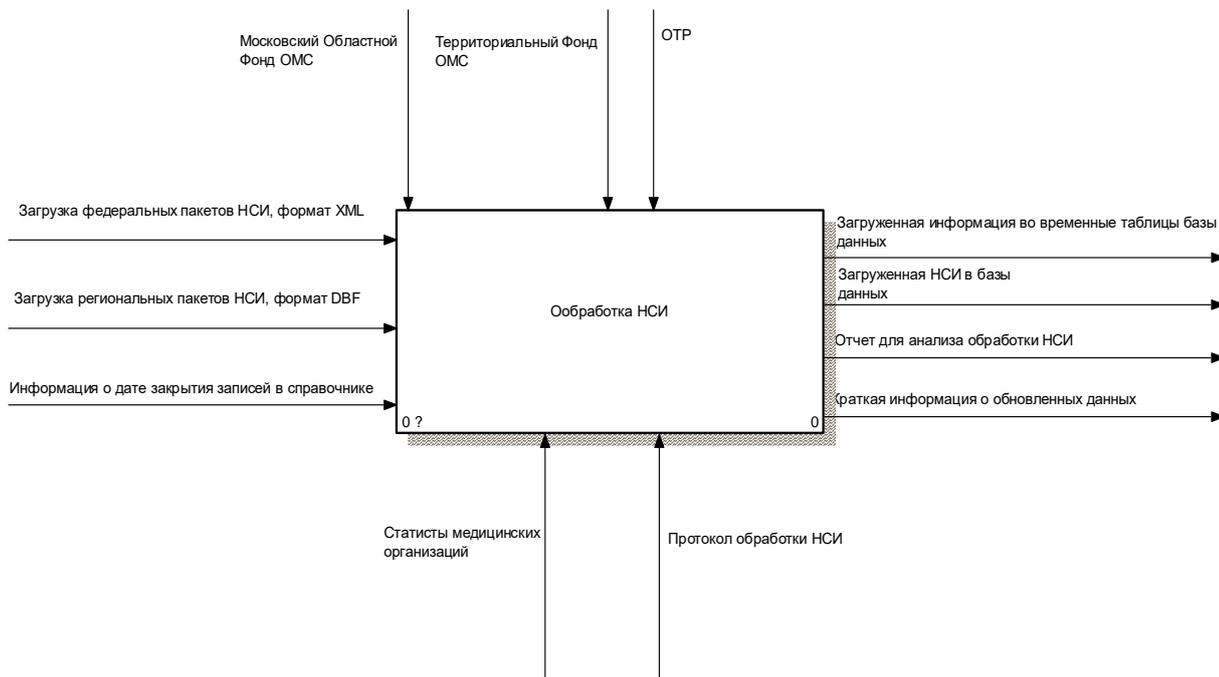


Рисунок 9 - Контекстная диаграмма «Обработка НСИ»

На вход поступает нормативно-справочная информация разных форматов, а также информация о дате закрытия записей в базе данных (далее БД).

Управляющими, регламентирующими и нормативными данными, которыми руководствуется работа протоколов, является, «ОТР», а также Территориальный фонд ОМС Московской области и Московский областной фонд ОМС.

Механизмами являются статисты медицинских организаций и непосредственно разработанный протокол обработки НСИ.

На выход поступает информация о изменениях справочников НСИ, информация, хранящаяся во временных таблицах, краткий отчет о изменениях и подробный отчет для анализа.

Декомпозиция контекстной диаграммы «Обработка НСИ» отображает три основных процесса (рисунок 10):

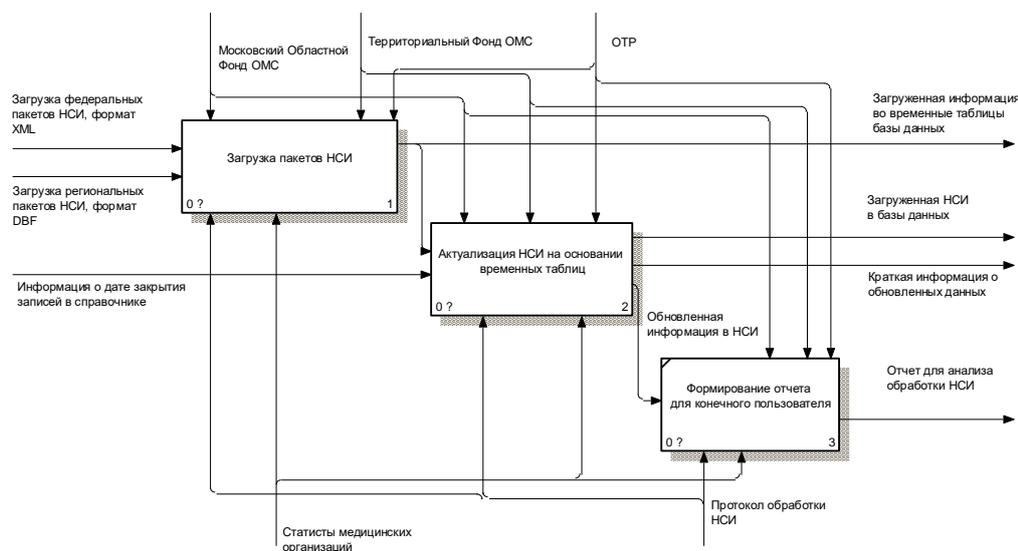


Рисунок 10 - Декомпозиция контекстной диаграммы «Обработка НСИ»

- 1) Загрузка пакета НСИ;
- 2) Актуализация НСИ на основании временных таблиц;
- 3) Формирование отчета для конечного пользователя.

В блоке «Загрузка пакета НСИ» входными данными являются пакеты НСИ разного формата (региональные и территориальные справочники имеют различные форматы). На выходе мы получаем подготовленные временные таблицы для дальнейшей обработки справочников в БД.

В блоке «Актуализация НСИ» входными данными являются подготовленные временные таблицы, а выходом является уже обновленная информация в НСИ.

Для блока «Формирование отчета для конечного пользователя» входными данными является обновлённая НСИ, а результат – файл для анализа изменений в нормативно-справочной информации.

Для декомпозиции блока «Загрузка пакетов НСИ» входными параметрами являются пакеты НСИ, разных форматов (МОФОМС – XML, ТФОМС – DBF), а результатом – загруженная и уже обработанная информация во временные таблицы, для дальнейшего преобразования справочников базы данных. Каждые новые данные, стирают предыдущие во временных таблицах, данный процесс служит для оптимизации места в БД.

Декомпозиция блока «Загрузка пакетов НСИ» представлена на рисунке 11.

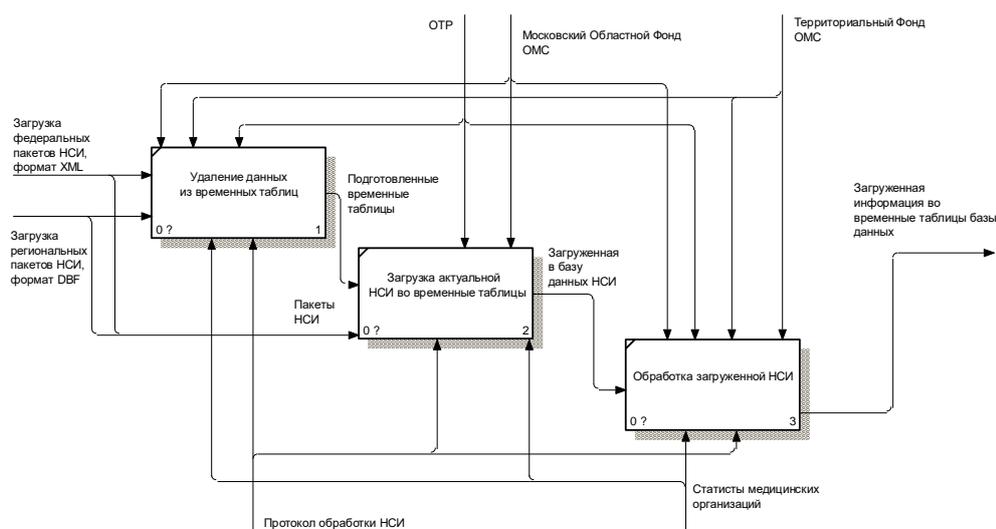


Рисунок 11 - Декомпозиция блока «Загрузка пакета НСИ»

Декомпозиция блока «Актуализация НСИ» была реализована в нотации IDEF3 (рисунок 12), которая предназначается для описания бизнес-процессов нижнего уровня.

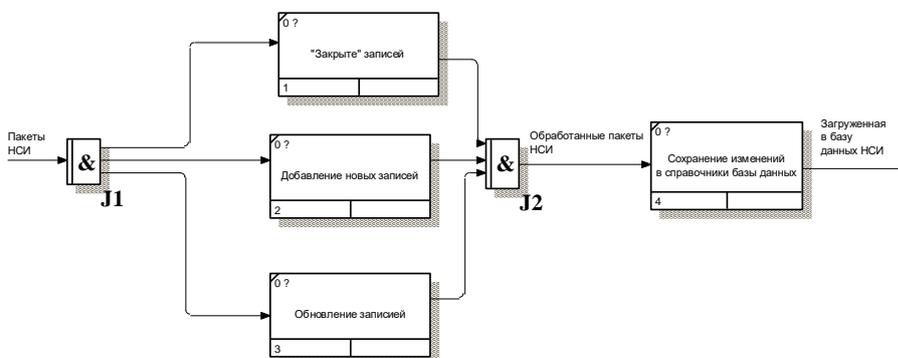


Рисунок 12 - Декомпозиция блока «Актуализация НСИ»

Декомпозиция на рисунке 9 содержит объекты – логические операторы, с помощью которых показывают альтернативы и места принятия решений, а также объекты – стрелки для отображения последовательности работ в бизнес-процессе.

На рисунке 13 представлена контекстная диаграмма бизнес-процесса протокола выгрузки реестров счетов.

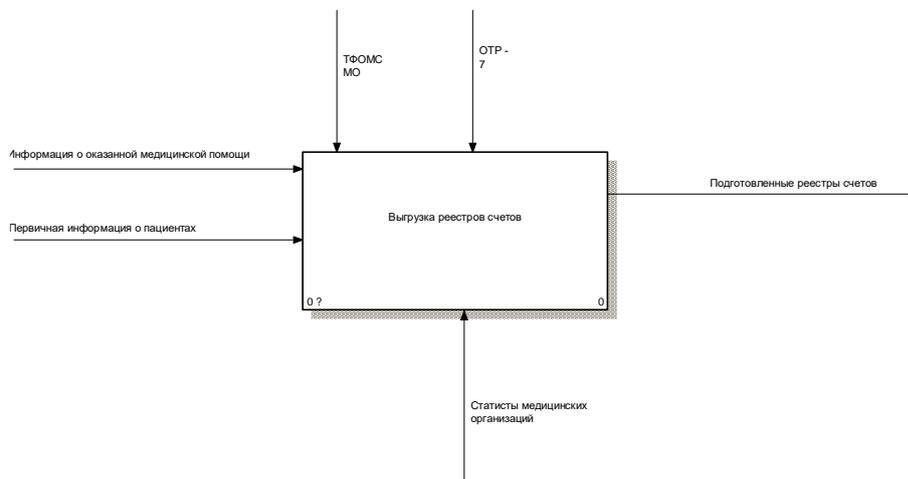


Рисунок 13 - Декомпозиция блока «Актуализация НСИ»

Данная модель на текущем этапе не отличается от контекстной диаграммы бизнес-процесса протокола выгрузки реестров счетов «как есть». Декомпозиция блока «Актуализация НСИ» - рисунок 14.

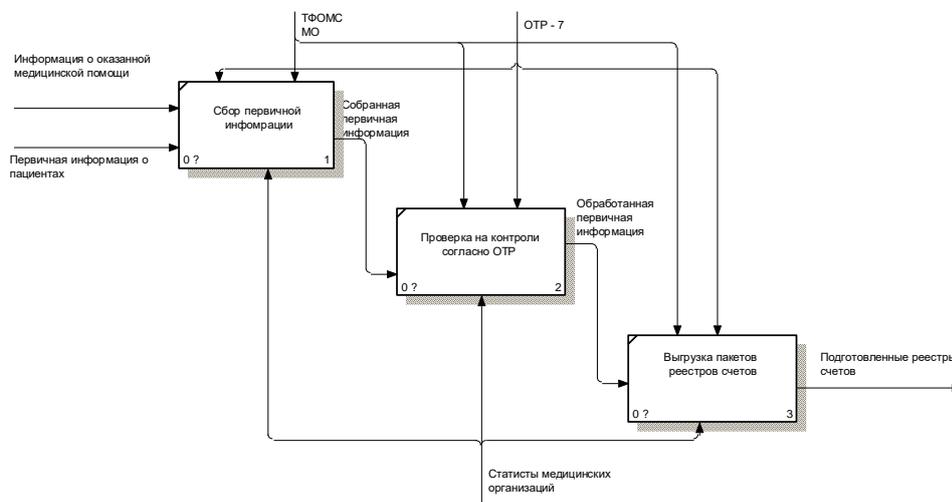


Рисунок 14 - Декомпозиция блока «Актуализация НСИ»

В отличии блока «Актуализация НСИ» был реализован процесс проверки на контроли согласна ОТР, что приведет к более корректному оформлению случаев и за счет автоматизированной обработки – сокращению потраченного времени.

## 3.2 Структура базы данных

Для построения информационной модели была выбрана программа ERwin.

ERWin - это не просто средство проектирования, но и инструмент разработки, способный автоматически создавать таблицы и генерировать текст хранимых процедур для всех популярных СУБД. Была построена ER-диаграмма физического уровня, на которой были выделены сущности, их атрибуты, первичные ключи, идентифицированы связи между сущностями.

Физический уровень модели составляют целевая СУБД, имена объектов и типы данных, индексы.

Основными таблицами являются:

- таблица талонов (hlt\_tap);
- таблица оказанных медицинских услуг (hlt\_smtap);
- таблица медицинских карт (hlt\_mkab);
- таблица докторов (hlt\_lpudoctor);
- таблица ЛПУ (oms\_lpu);
- таблица регистр оказанных услуг (oms\_servicemedical);
- таблица тарифов (oms\_tariff);
- таблицы диагнозов (oms\_mkb);
- таблицы, содержащие нормативно-справочную информацию (специальности, профили отделений, типы отделений и др.).

Для оформления корректной первичной информации, талоны амбулаторных пациентов должны содержать в себе максимальную информацию о оказанной услуге. Поэтому, данная таблица имеет множество связей.

Таблицы онко-справочников имеют тесную связь между собой, и при выборе значения в одной таблицы, варианты выбора другой будут усекаются по причине данных взаимосвязей, благодаря чему, у врачей пропадает необходимость фильтровать ненужные записи, что естественно облегчает работу, т.к. справочники могут достигать размеров более двух тысяч строк.

Структура базы данных представлена на рисунках 15.

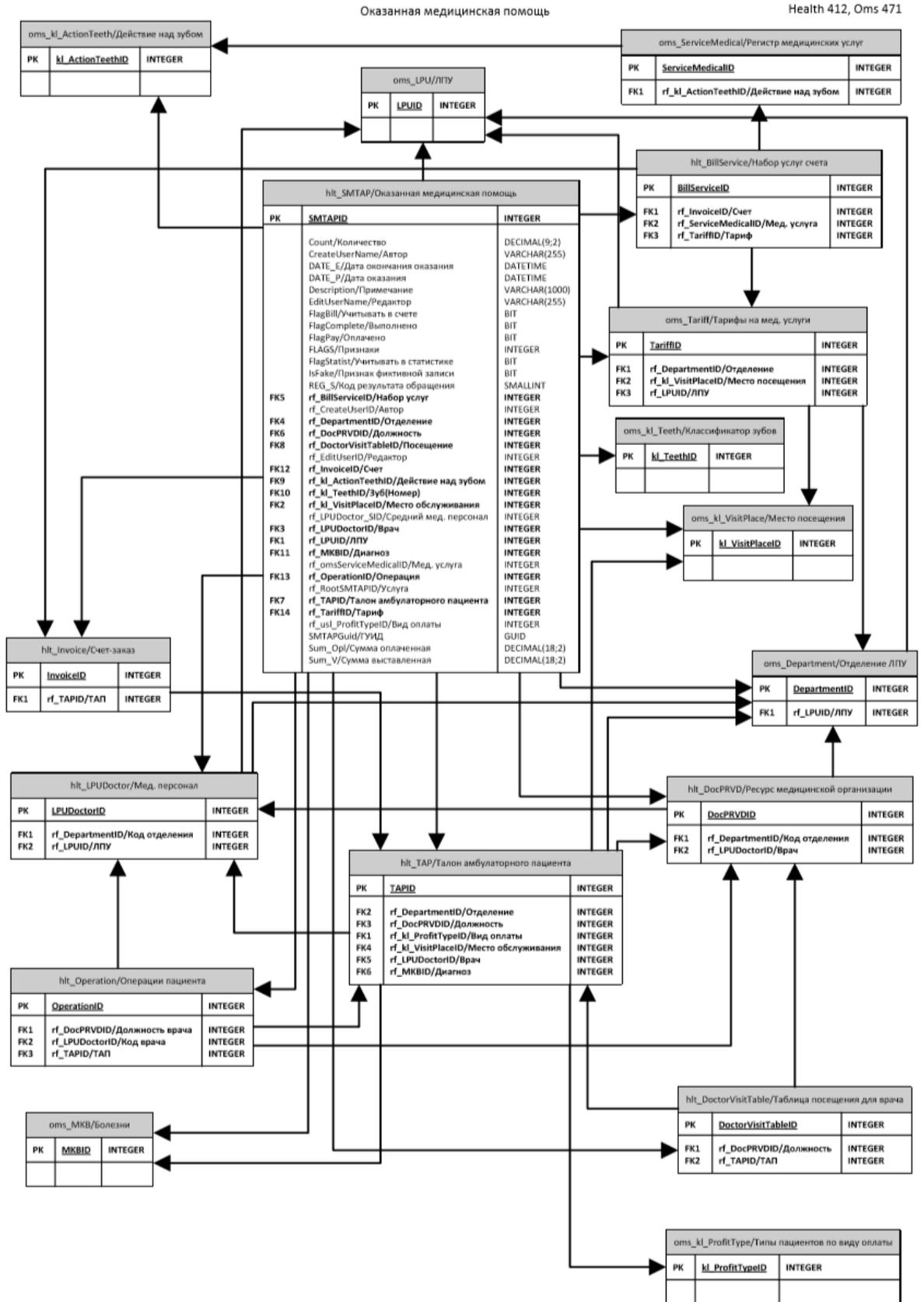


Рисунок 15 – Структура базы данных

Компания «СофтТраст» с каждым годом увеличивает количество разрабатываемых модулей, поэтому структура базы данных в дальнейшем будет расти.

### 3.3 Пример работы программного обеспечения

Так как МИС содержит в себе 2 режима работы (администрирование и рабочий режим), разработчиками было переустроены разные права доступа работы с МИС. По этой причине, для каждого юзера определяются:

- режим работы в МИС: «Windows авторизация», а также «Общая авторизация»;
- в режиме «Общая авторизация» пользователь обязан ввести его логин и пароль.

Форма авторизации с выбранным режимом работы - «Windows авторизация» представлен на рисунке 16.

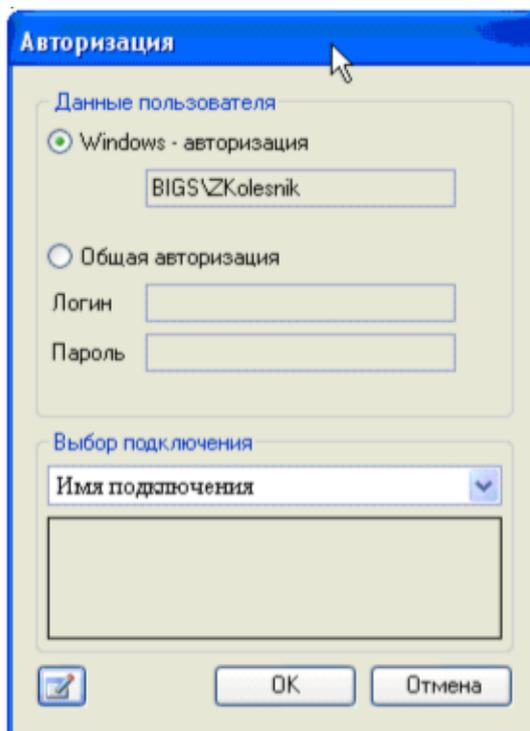


Рисунок 16 – Форма авторизации пользователя

При выборе работы «Windows-авторизация», из названия понятно, что данные пользователя – имя и пароль системы Windows. Данные определяются администратором локальной сети или компьютера.

В режиме «Общая авторизация» для входа в МИС пользователю обязан заполнить поля: «Логин» и «Пароль». Данные пользователя назначаются администратором МИС.

Во время входа в МИС, пользователю определяются его доступные функции в МИС.

В следствии удачной авторизации на экран выводится главное окно изображенное на рисунке 17.

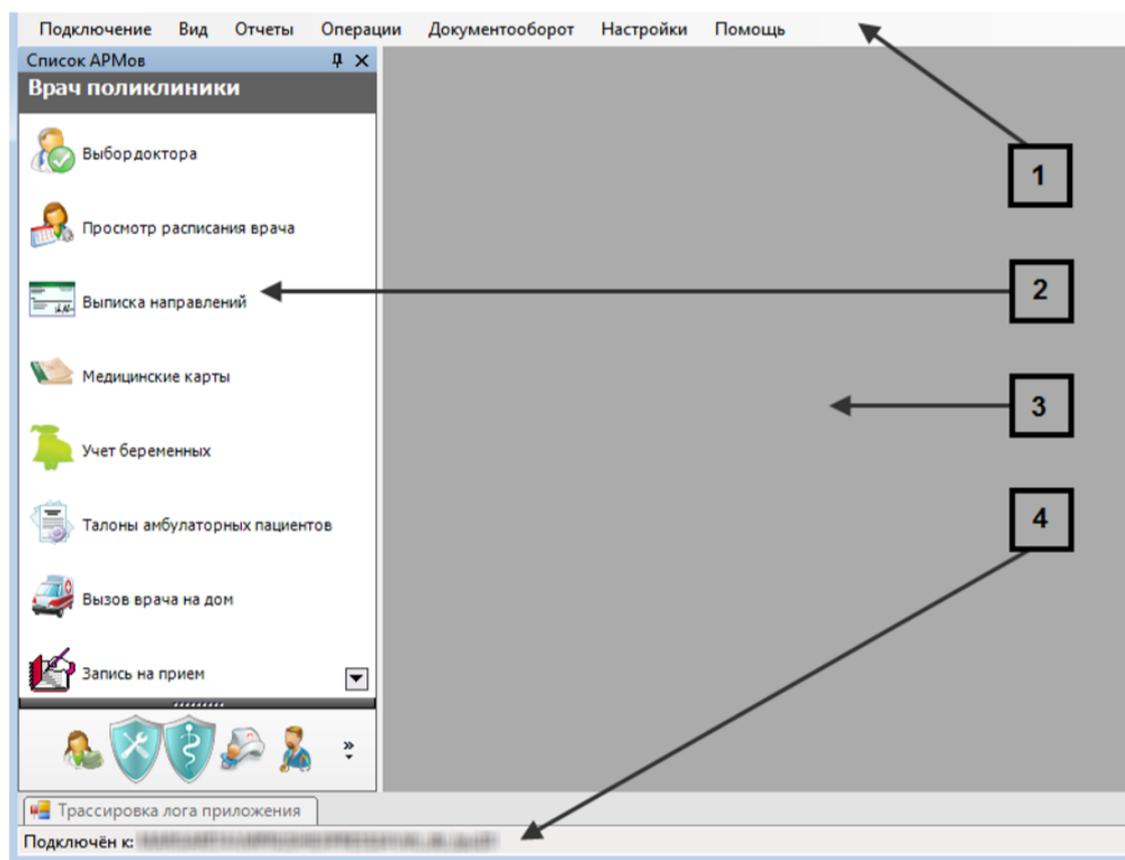


Рисунок 17 – Главное окно МИС

Главное окно МИС содержит основные 4 области работы:

- элементы управления;
- автоматизированное рабочее место;

- основная рабочая область;
- строка подключения.

Область «Элементы управления» предназначена для выполнения основных функциональных действий (вызов помощи, вход под другим именем пользователя и другие).

Область «Автоматизированное рабочее место» (АРМ) служит для быстрого перехода между АРМ.

«Основная рабочая область» содержит в себе основные формы, предназначенные для работы пользователя.

«Строка подключения» - информативная область, предназначена для отображения к какой базе подключен клиент, состояние подключения, а также, название текущей экранной формы.

Следует также подробно остановиться на описании рабочей области МИС. Данная область для любого документа отображает унифицированную форму (рисунок 18), для работы с выбранным документов.

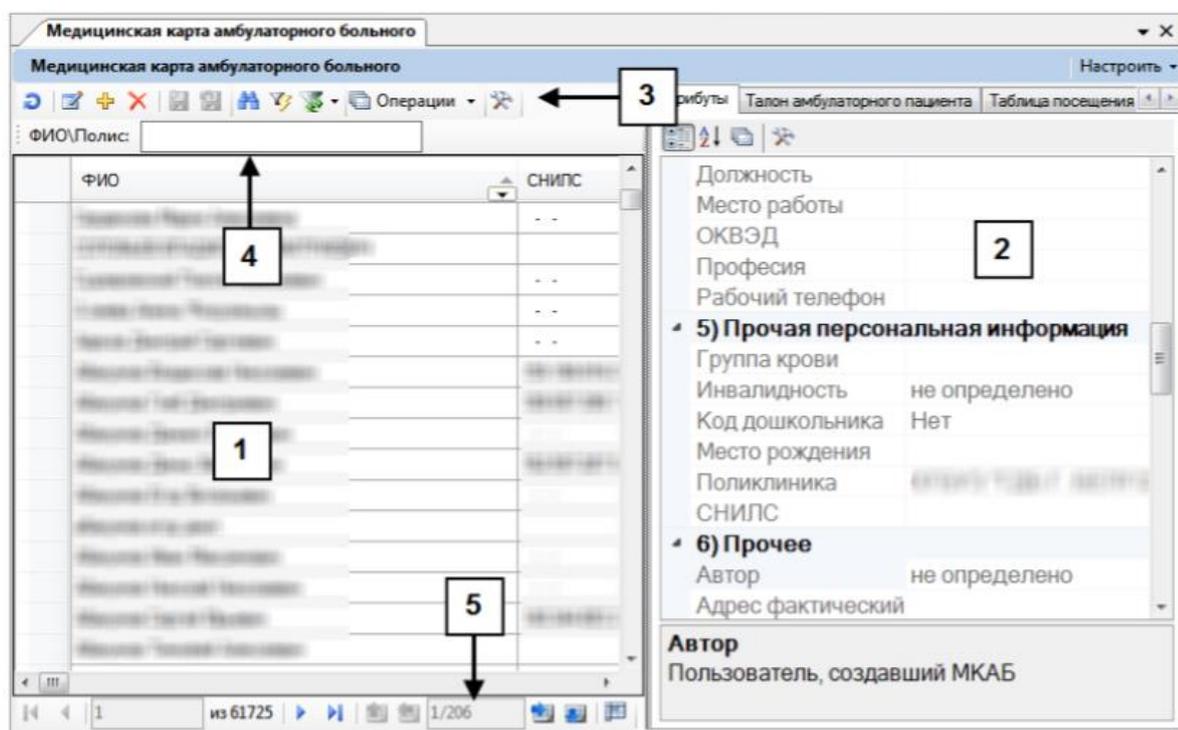


Рисунок 18 – «Основная рабочая область»

«Основная рабочая область» состоит из 5 рабочих областей:

1. Данные документов;
2. Быстрый вывод информации по выбранной записи.
3. Элементы управления;
4. Панель быстрого поиска информации;
5. Панель навигации.

Пример работы окна МИС в режиме администрирования, на вкладке протоколы представлен на рисунке 19.

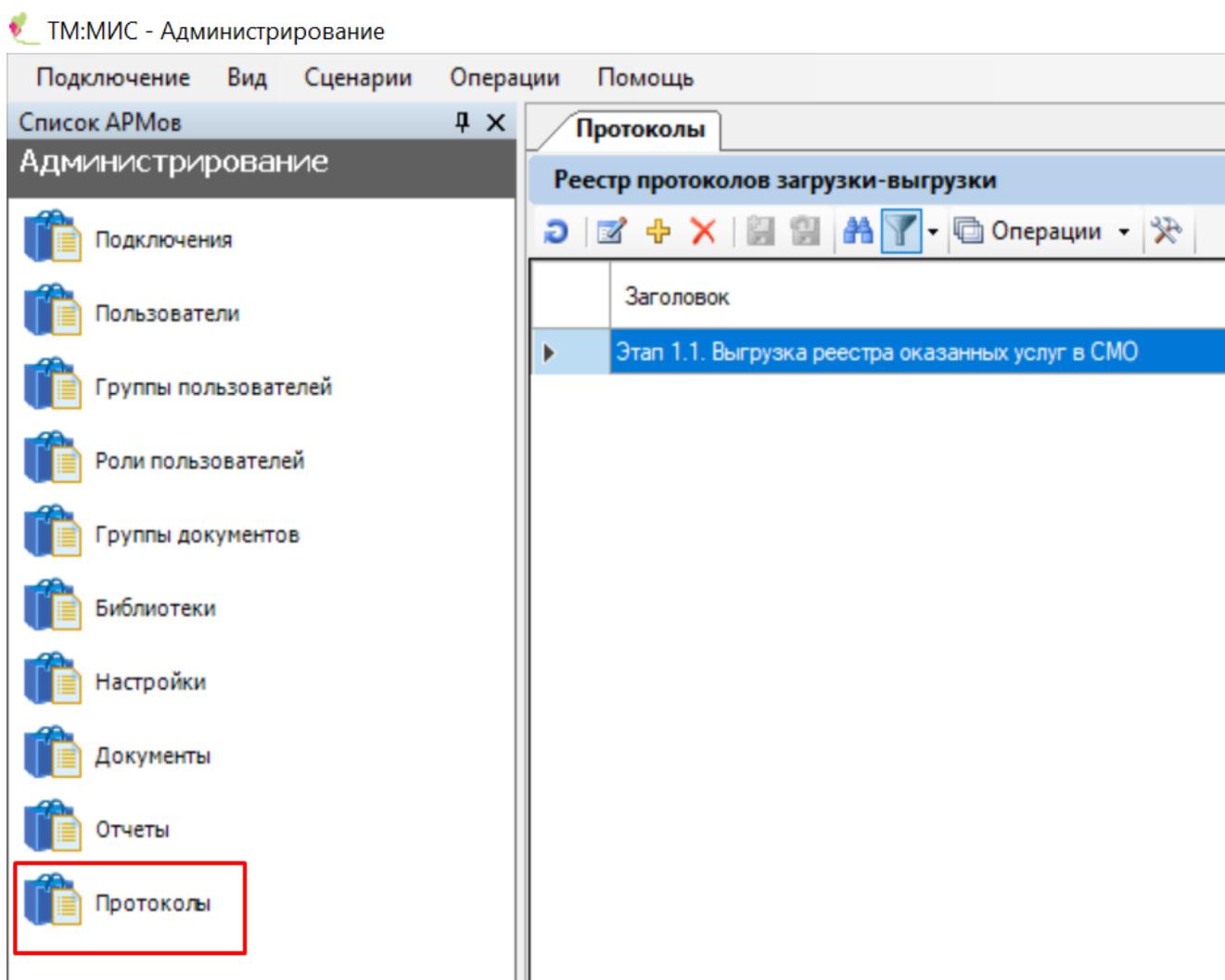


Рисунок 19 – Пример работы в режиме администрирование

В данной вкладке разработчики создают/редактируют протоколы, отчеты и пользуются другими функциями доступными для администраторов.

Модуль «Взаиморасчеты» представлен на рисунке 20. В данном модуле, работники медицинских организация проводят выгрузку реестров счетов. В качестве примера представлен реестр за май 2019 года.

The screenshot shows the 'Взаиморасчеты' module interface. At the top, there are filters for the reporting period (May 2019) and payment type (OMS). Below this is a list of error codes and descriptions. A summary table is displayed with the following columns: Код ошибки, Описание, Амбулаторные карт, Случаев амбулатории, Сумма амбулаторных случаев, Пациентов стационара, Стационарных случаев, Сумма стационарных случаев, Пациентов дневного стационара, Случаев дневного стационара.

Код ошибки	Описание	Амбулаторные карт	Случаев амбулатории	Сумма амбулаторных случаев	Пациентов стационара	Стационарных случаев	Сумма стационарных случаев	Пациентов дневного стационара	Случаев дневного стационара
	Всего ошибок	2958	13447	11781202	122	123	3241978	152	16
	Всего по счету	9553	15730	15669872	544	545	12786001	623	62
-1 5702	Штраф Дублирование КСГ/КПГ в случае	0	0	0	0	0	0	0	2
-1 ДарAge	Возраст не соотв услуге для случаев диспансеро...	11	11	10124	0	0	0	0	0
-1 ДарDu1	Дубли случаев диспансеризации и мед.осмотров в текущем...	3	6	4550	0	0	0	0	0
-1 ДарPer	Штраф Выставленные в прошлых периодах случаи диспансер...	109	109	1096002	0	0	0	0	0
-1 ДарSex	Пол не соотв услуге для случаев диспансеризации и мед.ос...	52	53	43351	0	0	0	0	0
-1 DS10	Штраф Диагноз не оплачивается по ОМС	281	297	248505	26	26	949598	2	0
-1 DS11	Штраф Диагноз Z выставлен для посещений по заболеванию	638	746	336444	0	0	0	0	0
-1 DS13	Стоматологические услуги не соответствуют диагнозу	40	55	16164	0	0	0	0	0
-1 Sex1	Отчество пациента не соответствует полу	31	52	40905	0	0	0	0	1
-1 Z001	Штраф Установлен диагнос Z001 для детей от 3 лет	44	45	23414	0	0	0	0	0
-1 Z002	Штраф Установлен диагнос Z002 для детей менее 3 лет и б...	16	17	8066	0	0	0	0	0
-1 Z003	Штраф Установлен диагнос Z003 для детей менее 15 лет и ...	39	39	21257	0	0	0	0	0
-1 АнВых	Анб случай оказан в воскресенье или в праздничный день	178	181	296177	0	0	0	0	0
-1 ДейстУс	ФЛК Указана не действующая услуга	147	147	116679	0	0	0	0	0
-1 ЗубД	Дубли зубной формулы по услугам	5	5	4884	0	0	0	0	0
-1 ЗубСТ	Ошибка ввода зубной формулы	975	1564	579119	0	0	0	0	0
-1 СтДубл	Дубли первичных стоматологических приемов	125	278	50629	0	0	0	0	0
-1 СтУсл	Не верно указана стоматологическая услуга	2589	3009	2318159	0	0	0	0	0
-1 тариф	Услуга не по спецтарифу	60	61	23112	0	0	0	0	0
-1 ТМ_УСВ	Штраф_Услуга не соответствует возрасту	4520	6955	4946484	0	0	0	0	0

Рисунок 20 – Модуль «Взаиморасчеты»

На модуле взаиморасчетов отображена информация о пациентах в разрезе страховых, перечень потенциальных ошибок по случаям (вкладка экспертиза), быстрый переход на карту больного (вкладка сведения об оказанной медицинской помощи). Также имеется возможность загружать форматно-логический контроль (список ошибок от страховых) и напрямую работать с перечнем из данного файла.

Пользователи, посредством клиентского приложения, отправляют на сервер запросы или вызывают хранимые на сервере процедуры. В ответ сервер выполняет необходимые операции с данными и отправляет результаты своей работы клиентам.

В данном разделе выпускной квалификационной работы были разработаны функциональная и информационная модель обработки НСИ и

выгрузки реестров счетов для медицинской организации МО. А также рассмотрены и выбраны средства реализации для визуального интерфейса и серверной части.

### 3.4 Описание контрольного примера реализации проекта

Далее подробно описана разработка протоколов обработки нормативно-справочной информации и выгрузки реестров счетов для медицинской организации Московской области.

Для разработки протокола обработки НСИ первым делом необходимо написать скрипт заполнения временных таблиц из источников.

Федеральные справочники даются в формате XML, для их загрузки во временные таблицы используются XSD-схема загрузки. Пример схемы загрузки для справочника «Классификатор типов лучевой терапии» (N017.XML) представлен на рисунке 21.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:sql="urn:schemas-microsoft-com:mapping-schema">
  <xs:annotation>
    <xs:documentation>"N017.xml"</xs:documentation>
  </xs:annotation>
  <xs:element name="zap" sql:relation="t_N017">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="ID_TLuch" />
        <xs:element type="xs:string" name="TLuch_NAME" />
        <xs:element type="xs:string" name="DATEBEG" />
        <xs:element type="xs:string" name="DATEEND" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Рисунок 21 – XSD-схема загрузки

Для успешной загрузки необходимо на вкладке «Таблицы» указать наименование временной таблицы и такое же название продублировать в XSD-схеме загрузки.

В рамках поставленной задачи была реализована массовая загрузка федеральных справочников. Загрузка всех справочников должна быть строго

одного формата. Список загружаемых федеральных справочников представлен на рисунке 22.

Формирование источников данных	
Загрузка из XML	tmp_t_out_t_F008
Загрузка из XML	tmp_t_out_t_F009
Загрузка из XML	tmp_t_out_t_F011
Загрузка из XML	tmp_t_out_t_V002
Загрузка из XML	tmp_t_out_t_V006
Загрузка из XML	tmp_t_out_t_V008
Загрузка из XML	tmp_t_out_t_V009
Загрузка из XML	tmp_t_out_t_V010
Загрузка из XML	tmp_t_out_t_V012
Загрузка из XML	tmp_t_out_t_V014
Загрузка из XML	tmp_t_out_t_V020
Загрузка из XML	tmp_t_out_t_V021
Загрузка из XML	tmp_t_out_t_V023
Загрузка из XML	tmp_t_out_t_V024
Загрузка из XML	tmp_t_out_t_V025
Загрузка из XML	tmp_t_out_t_V026
Загрузка из XML	tmp_t_out_t_N001
Загрузка из XML	tmp_t_out_t_N002
Загрузка из XML	tmp_t_out_t_N003
Загрузка из XML	tmp_t_out_t_N004
Загрузка из XML	tmp_t_out_t_N005
Загрузка из XML	tmp_t_out_t_N006
Загрузка из XML	tmp_t_out_t_N007
Загрузка из XML	tmp_t_out_t_N008
Загрузка из XML	tmp_t_out_t_N009
Загрузка из XML	tmp_t_out_t_N010
Загрузка из XML	tmp_t_out_t_N011
Загрузка из XML	tmp_t_out_t_N012
Загрузка из XML	tmp_t_out_t_N013
Загрузка из XML	tmp_t_out_t_N014
Загрузка из XML	tmp_t_out_t_N015
Загрузка из XML	tmp_t_out_t_N016
Загрузка из XML	tmp_t_out_t_N017
Загрузка из XML	tmp_t_out_t_N018
Загрузка из XML	tmp_t_out_t_N019

Рисунок 22 – Список загружаемых федеральных справочников

Также, перед загрузкой НСИ следует учитывать, что в БД для корректной работы ТМ:МИС всегда должна быть «нулевая строка», которая не является активной для выбора пользователем. Поэтому, необходимо было это учесть и изменить значения кодов в «нулевой строке», что и было сделано. Данное изменение должно выполняться на первом этапе.

Предварительный скрипт для изменения «нулевой строки» представлен на рисунке 23.

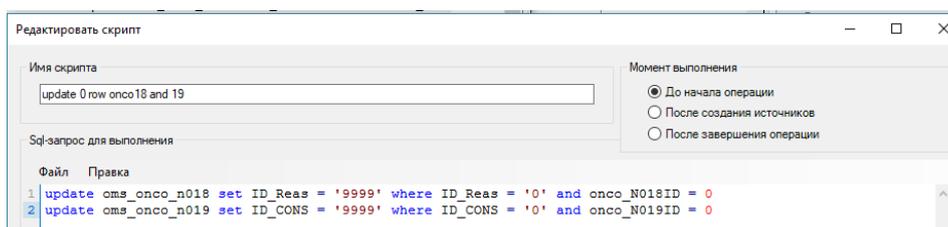


Рисунок 23 – скрипт для изменения «нулевой строки»

Следующим этапом протокол должен обрабатывать записи в таблицах, а именно изменять дату действия в случаях:

- запись отсутствует в справочниках;
- запись, не действующая в справочниках.

Пример реализации такого скрипта справочника «классификатор типов лучевой терапии» (N017.XML) представлен на рисунке 24.

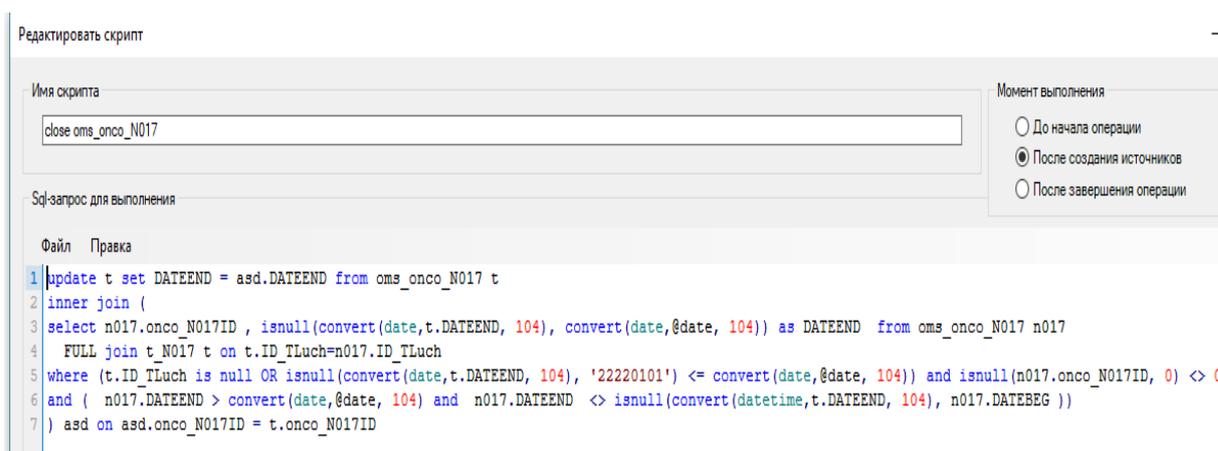


Рисунок 24 – Скрипт на закрытие записей

Аналогично загрузке, данный скрипт должен быть разработан на каждую таблицу и проводится обязательно после загрузки временных таблиц, так как в данном случае уже происходит проверка данных в БД и справочников.

Список изменяемых справочников представлен на рисунке 25.

Выполнение скриптов	
close oms_mkb	update t set Date_E = asd.d_e from oms_mkb t inner join ...
close oms_onco_N004	/*update oms_onco_N004 set DATEEND = DateBeg wh...
close oms_onco_N005	/*update oms_onco_N005 set DATEEND = DateBeg wh...
close oms_onco_N006	/*update oms_onco_N006 set DATEEND = DateBeg wh...
close oms_onco_N007	/*update oms_onco_N007 set DATEEND = DateBeg wh...
close oms_onco_N008	/*update oms_onco_N008 set DATEEND = DateBeg wh...
close oms_onco_N009	/*update oms_onco_N009 set DATEEND = DateBeg wh...
close oms_onco_N010	/*update oms_onco_N010 set DATEEND = DateBeg wh...
close oms_onco_N011	/*update oms_onco_N011 set DATEEND = DateBeg wh...
close oms_onco_N012	/*update oms_onco_N012 set DATEEND = DateBeg wh...
close oms_onco_N013	/*update oms_onco_N013 set DATEEND = DateBeg wh...
close oms_onco_N014	/*update oms_onco_N014 set DATEEND = DateBeg wh...
close oms_onco_N015	/*update oms_onco_N015 set DATEEND = DateBeg wh...
close oms_onco_N016	/*update oms_onco_N016 set DATEEND = DateBeg wh...
close oms_onco_N017	/*update oms_onco_N017 set DATEEND = DateBeg wh...
close oms_onco_N018	/*update oms_onco_N018 set Date_E = Date_B where o...
Close oms_onco_N019	/*update oms_onco_N019 set Date_E = Date_B where o...
close oms_kl_TipOMS	/* update oms_kl_TipOMS set Date_E = Date_B where ...
close oms_kl_SocStatus	/*update oms_kl_SocStatus set Date_E = Date_B where ...
close oms_TYPEDOC	/*update oms_TYPEDOC set Date_E = Date_B where T...
close oms_kl_DepartmentProfile	/*update oms_kl_DepartmentProfile set Date_E = Date_B...
close oms_kl_DepartmentType	/*update oms_kl_DepartmentType set Date_E = Date_B ...
close oms_kl_MedCareType	/*update oms_kl_MedCareType set Date_E = Date_B wh...
close oms_ServiceMedical	/*update t set Date_E = asd.DATEEND from oms_Servic...
update oms_kl_MedCareUnit.CODE	update mcu set CODE=cast(code as int) from oms_kl_Me...
close oms_kl_MedCareUnit	/*update oms_kl_MedCareUnit set Date_E = Date_B whe...
close oms_kl_VisitResult	/*update oms_kl_VisitResult set Date_E = Date_B where ...
close oms_kl_StatCureResult	/*update oms_kl_StatCureResult set Date_E = Date_B w...
update stt_EmerSign	update EmerS set CODE=cast(code as int) from stt_Emer...
close stt_EmerSign	/*update stt_EmerSign set Date_E = Date_B where Emer...
close stt_BedProfile	/*update stt_BedProfile set Date_E = Date_B where Bed...
close oms_PRVS	/*update oms_PRVS set Date_End = Date_BEG where ...
close oms_kl_ReasonType	/*update oms_kl_ReasonType set Date_E = Date_B wh...
close oms_Tariff	update oms_Tariff set Date_E = Date_B where tariffid in (...)
close oms_onco_N001	/*update oms_onco_N001 set DATEEND = DateBeg wh...
close oms_onco_N002	/*update oms_onco_N002 set DATEEND = DateBeg wh...
close oms_onco_N003	/*update oms_onco_N003 set DATEEND = DateBeg wh...

Рисунок 25 – Список изменяемых справочников

Следующим этапом необходимо было создать скрипты на добавление новых записей.

Листинг 1 содержит скрипт добавления записей справочника «классификатор типов лучевой терапии» (N017.XML).

Листинг 1 - добавление записей справочника «классификатор типов лучевой терапии»

```
select t.ID_TLuch,  
       left(t.TLuch_NAME, 255) as TLuch_NAME,  
       CONVERT(date,t.DATEBEG, 104 )as DATEBEG,  
       CONVERT(date,isnull(t.DATEEND,'22220101'), 104 ) as DATEEND  
from oms_onco_N017 n017  
full join t_N017 t on t.ID_TLuch=n017.ID_TLuch  
where isnull(n017.onco_N017ID, -1) <> 0 and n017.onco_N017ID is null
```

Листинг 2 содержит скрипт обновления данных справочника «классификатор типов лучевой терапии» (N017.XML).

Листинг 2 - обновление данных справочника «классификатор типов лучевой терапии»

```
select t.ID_TLuch,  
       left(t.TLuch_NAME, 255) as TLuch_NAME,  
       CONVERT(date,t.DATEBEG, 104 )as DATEBEG,  
       CONVERT(date,isnull(t.DATEEND,'22220101'), 104 ) as DATEEND  
from oms_onco_N017 n017  
inner join t_N017 t on t.ID_TLuch=n017.ID_TLuch  
where isnull(n017.onco_N017ID, -1) <> 0  
and CONVERT(date,isnull(t.DATEEND,'22220101'), 104 ) >  
convert(date,@date, 104)  
and  
(CONVERT(date,isnull(t.DATEEND,'22220101'), 104 ) <> n017.DATEEND  
or CONVERT(date,(t.DATEBEG), 104 ) <> n017.DATEBEG  
or left(t.TLuch_NAME, 255) <> n017.TLuch_NAME)
```

Разработка протокола выгрузки реестров счетов начинается с настройки проставления номера пакета (Листинг 3).

Листинг 3 - скрипт настройка для номера пакета

```
declare @new_num varchar(100),@num varchar(100),@old_num varchar(100);

Set @num=case when '@TEST'='0' then /*проверочный*/
'REESTRTEST_'+@FCOD+'50'+@mm+'@yy'
else /*контрольный*/'REESTR_'+@FCOD+'50'+@mm+'@yy' end

if not exists ( select top 1 ValueStr from x_UserSettings
                where [Property] = 'Идентификатор пакета СМО МО')

begin

/*Если нет настройки, то добавляем*/

insert into x_UserSettings
(ValueStr,[Property],rf_UserID,OwnerGUID,DocTypeDefGUID,rf_SettingTypeID)

Select @num+'1','Идентификатор пакета СМО МО',0,'00000000-0000-0000-0000-
000000000000', '00000000-0000-0000-0000-000000000000',0

End else

Begin /*Если есть настройка, то необходимо определить тип реестра*/

Select @old_num=ValueStr

from x_UserSettings

where [Property] = 'Идентификатор пакета СМО МО'

Select @new_num=case when dbo.IsDigits(Replace(@old_num,@num,''))>0

then @num+cast(dbo.IsDigits(Replace(@old_num,@num,''))/*+1*/ as varchar) else
@num+'1' end
```

```
Update x_UserSettings set ValueStr=@new_num where [Property] =  
'Идентификатор пакета СМО МО'
```

End

Следующий этап реализация таблиц реестров. Свод данных состоит из 12 таблиц. Перечень таблиц представлен на рисунке 26.

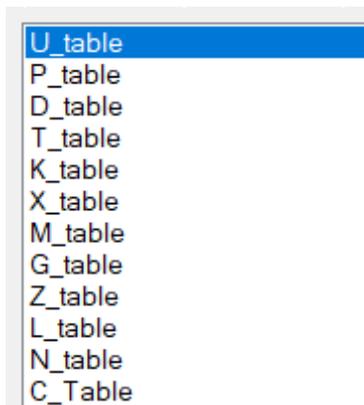


Рисунок 26 – Перечень вспомогательных таблиц для выгрузки реестров счетов

Завершающий этап – выгрузка созданных таблиц в файлы dbf. Имя файла выгрузки формируется из для каждой ЛПУ уникально. Листинг 4 - формирования имени.

Листинг 4 - формирования имени.

```
Reestr\TFOMS_1\[Sql select top 1 ValueStr from x_UserSettings where  
Property = 'Идентификатор пакета СМО МО']\[Sql SELECT '@DirName']\[Sql  
SELECT 'P'+@FileName'].DBF
```

### 3.5 Тестирование и отладка программного продукта

Тестирование программного обеспечения – процесс исследования программного продукта, с целью установления соответствия между реальным поведением программы и её ожидаемым поведением на конечном наборе тестов.

Для начала был протестирован протокол обработки нормативно-справочной информации. В качестве примера был выбран справочник - N020 Классификатор лекарственных препаратов, применяемых при проведении лекарственной терапии (OnkLekp) (рисунок 27).

```
N0201218.xml — Блокнот
Файл  Правка  Формат  Вид  Справка
<zap>
  <ID_LEKP>002463</ID_LEKP>
  <MNN>Эхинацеи пурпурной травы настойка</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002464</ID_LEKP>
  <MNN>Эхинацеи пурпурной травы сок</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002465</ID_LEKP>
  <MNN>Эхинацеи пурпурной травы экстракт</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002466</ID_LEKP>
  <MNN>Эхинацеи узколистной настойка</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002467</ID_LEKP>
  <MNN>Яблочная кислота</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002468</ID_LEKP>
  <MNN>Яд гадюки</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002469</ID_LEKP>
  <MNN>Якорцев стелющихся травы экстракт</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002470</ID_LEKP>
  <MNN>Янтарная кислота</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
<zap>
  <ID_LEKP>002471</ID_LEKP>
  <MNN>Отсутствие МНН у лекарственного препарата</MNN>
  <DATEBEG>01.01.2019</DATEBEG>
  <DATEEND/>
</zap>
</packet>
```

Рисунок 27 – Классификатор лекарственных препаратов, применяемых при проведении лекарственной терапии (OnkLekp)

Данный справочник содержит действующую 2471 запись.

Предварительно в БД данный справочник полностью отсутствовал, что продемонстрировано на рисунке 28.

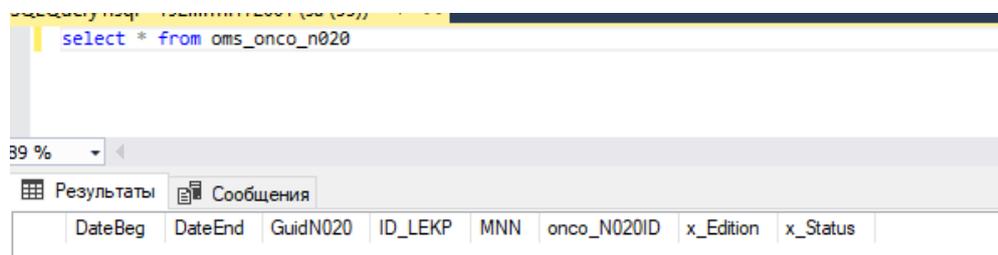


Рисунок 28 – Таблица oms\_onco\_n020 в БД до проведения протоколов

В качестве еще одного примера был выбран справочник - N001 Классификатор противопоказаний и отказов (OnkPrOt) (рисунок 29).



Рисунок 29 – Классификатор противопоказаний и отказов (OnkPrOt)

Данный справочник содержит 8 действующих записей. Для проверки работоспособности данного прокола, в рабочей базе данных была обновлена

добавлена лишняя запись. В результате справочник до запуска протокола принимает вид (рисунок 30):

DATEBEG	DATEEND	GUIDN001	ID_PrOt	onco_N001ID	PrOt_NAME	x_Edition	x_Status
1900-01-01 00:00:00.000	2222-01-01 00:00:00.000	00000000-0000-0000-0000-000000000000	0	0		1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	ED16CBDC-7879-450A-96D2-A508C8FCAD5F	1	1	Противопоказания к проведению хирургического лече...	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	F847174A-3A67-4BD3-B177-4F270DAD5AB5	2	2	Противопоказания к проведению химиотерапевтическ...	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	DA58E6B4-8C42-4F8A-A325-D7CBCEEE2A58	3	3	Противопоказания к проведению лучевой терапии	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	569698B5-16FA-4A95-99BC-DD4BD2809450	4	4	Отказ от проведения хирургического лечения	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	4A62F915-E4D9-4E40-836F-C653BF22FF45	5	5	Отказ от проведения химиотерапевтического лечения	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	998F98EE-F8AA-4FC0-8A50-7C6854B5D1F9	6	6	Отказ от проведения лучевой терапии	1	1
1900-01-01 00:00:00.000	2222-01-01 00:00:00.000	0B6CB308-EDC3-4D16-98B5-207B7BC5CE17	10	7	тест	1	1

Рисунок 30 – Таблица oms\_onco\_n001 в БД до проведения протоколов

При запуске протокола обработки НСИ необходимо будет указать необходимые справочники.

Загрузка всех N файлов представлена на рисунке 31.

Имя файла	Путь к файлу
N0011118.xml	C:\Users\ARyshkova\Desktop\N
N0021218.xml	C:\Users\ARyshkova\Desktop\N
N0031218.xml	C:\Users\ARyshkova\Desktop\N
N0041218.xml	C:\Users\ARyshkova\Desktop\N
N0050119.xml	C:\Users\ARyshkova\Desktop\N
N0061218.xml	C:\Users\ARyshkova\Desktop\N
N0070818.xml	C:\Users\ARyshkova\Desktop\N
N0080918.xml	C:\Users\ARyshkova\Desktop\N
N0090918.xml	C:\Users\ARyshkova\Desktop\N
N0100418.xml	C:\Users\ARyshkova\Desktop\N
N0111218.xml	C:\Users\ARyshkova\Desktop\N
N0120918.xml	C:\Users\ARyshkova\Desktop\N
N0131218.xml	C:\Users\ARyshkova\Desktop\N
N0141218.xml	C:\Users\ARyshkova\Desktop\N
N0150818.xml	C:\Users\ARyshkova\Desktop\N
N0160418.xml	C:\Users\ARyshkova\Desktop\N
N0171218.xml	C:\Users\ARyshkova\Desktop\N
N0181218.xml	C:\Users\ARyshkova\Desktop\N
N0191218.xml	C:\Users\ARyshkova\Desktop\N
N0201218.xml	C:\Users\ARyshkova\Desktop\N
N0210319.xml	C:\Users\ARyshkova\Desktop\N

Рисунок 31 – Загрузка всех N файлов

Для начала процесса обработки НСИ необходимо нажать «далее». Следующим этапом необходимо указать дату закрытия записей в БД.

Результат завершения обработки НСИ представлен на рисунке 32.

```

Общее время работы: 00:00:12
Загружается файл 'C:\Users\ARyshkova\Desktop\N\N017.xml'...
Загружается файл 'C:\Users\ARyshkova\Desktop\N\N018.xml'...
Загружается файл 'C:\Users\ARyshkova\Desktop\N\N019.xml'...
Загружается файл 'C:\Users\ARyshkova\Desktop\N\N020.xml'...
Загружается файл 'C:\Users\ARyshkova\Desktop\N\N021.xml'...
Обрабатываются приемники данных. Операция может занять продолжительное время.
2 записей было добавлено в таблицу [N001. Классификатор противопоказаний и отказов]
0 записей было обновлено в таблице [N001. Классификатор противопоказаний и отказов]
49 записей было добавлено в таблицу [N002. Классификатор стадий]
250 записей было обновлено в таблице [N002. Классификатор стадий]
78 записей было добавлено в таблицу [N003. Классификатор Tumor]
235 записей было обновлено в таблице [N003. Классификатор Tumor]
13 записей было добавлено в таблицу [N004. Классификатор Nodus]
175 записей было обновлено в таблице [N004. Классификатор Nodus]
6 записей было добавлено в таблицу [N005. Классификатор Metastasis]
106 записей было обновлено в таблице [N005. Классификатор Metastasis]
0 записей было добавлено в таблицу [N006. Справочник соответствия стадий TNM]
0 записей было обновлено в таблице [N006. Справочник соответствия стадий TNM]
0 записей было добавлено в таблицу [N007. Классификатор гистологии]
10 записей было обновлено в таблице [N007. Классификатор гистологии]
0 записей было добавлено в таблицу [N008. Классификатор результатов гистологии]
25 записей было обновлено в таблице [N008. Классификатор результатов гистологии]
0 записей было добавлено в таблицу [N009. Классификатор соответ. гистологии диагнозам]
24 записей было обновлено в таблице [N009. Классификатор соответ. гистологии]
    
```

Рисунок 32 – Результат завершения обработки НСИ

По результату выполнения протокола конечный пользователь видит краткий анализ изменений. Справочник N020 принимает вид, изображенный на рисунке 33.

DateBeg	DateEnd	GuidN020	ID_LEKP	MNN	onco_N020ID
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	5D495087-63B9-47D0-9967-0E9CA844E12B	000001	(2S)-2-Ацетамидопентадиновая кислота	1
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	DF96314C-82E9-4521-A86D-BC428B2538BA	000002	(SR, 6S) 7,8-Диэтидио-1-метил-4,5-эпоксиморфинан-3...	2
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	D8F6953C-4437-4FF6-A2ED-12233D4C564E	000003	(Пероциста весеннего корня+Тимьяна обыкновенно...	3
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	SF231852-1E61-4EF8-8139-D82564348D66	000004	[Календулы лекарственной цветки+Ромашки аптечной ...	4
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	394862EE-890D-4A46-8F88-A71B7493891B	000005	1,4-Бромбензил-2-хлорфенол	5
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	88908E9B-0A8C-4608-AF0A-B80414A1D77E	000006	1,3-Диметилбензилдиазолил триолоид	6
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	F39F1FC7-56C0-4884-A3F2-DA7B7C38075	000007	1,4-Диметил-7-(2-пропил)азулен	7
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	08D21589-3CA-495A-91E7-126C14405793	000008	1-[3-хлорфенил]бензилметилмочевина	8
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	8A9FEBA3-F078-43AE-BE40-9AC85D050693	000009	1-этилпиперидин-5-нитро-2-пропoxисбензол	9
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	58BC3467-D610-4076-A7EF-F1943579781	000010	1-Хлорметил-2,8,9-триокса-5-аза-1-силабцикло[3,3] у...	10
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	DA3FD883-9A4D-4857-8E52-781C63148A15	000011	2,4-диоксо-6-метил-1,2,3,4-тетрагидропиримидин	11
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	9D0D5A28-6394-4AF7-9F88-CFF083215156	000012	2-гидроксисубантиновая кислота	12
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	E78C48AD-28E0-422A-B445-4FB89F9F9229	000013	2-феноксизанол	13
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	88C7173B-99DF-4C8D-B6D9-968ED8BA9DCC	000014	4-амино-1-изобутил-1Н-имидазол[4,5-с]инолин	14
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	A0576612-8B2E-4910-9CB1-812FF99A7A04	000015	4-Нитро-N{(1RS)-1-(4-фторфенил)-2-(1-этилпиперидин-4-...	15
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	306E3380-AEF8-4794-8031-FE7DCC234359	000016	6-метил-2-этилпиперидин-3-ол соль (2S)-2-этиламинопент...	16
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	9B3A9553-A321-4996-A1DC-11C62524E944	000017	7-(Метилпиперидин-3-ил)-4-оксо-1,4-дигидро[1,2,4]триазоло[...	17
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	CDFF0833-512E-4AF1-8BC4-3D8981243987	000018	D,L-Гопантеновая кислота	18
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	43F35CC0-8DBE-4C66-B625-82247012FA3	000019	D,L-Трeо-1-пара-нитрофенил-2-дихлорэтиламино-про...	19
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	9D53F620-1569-4A9D-BE3C-0E789C8C9F4D	000020	L-аспарагиновая кислота	20
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	6CD678E4-4917-47E9-9056-483A1D044F08	000021	L-изопропил-L-глутамил-L-л-глутамат натрия соль	21
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	EAB8C4FB-83AE-4AAB-91EA-7E17F306CF45	000022	N{(1RS,2RS)-2-гидроксис-1-гидроксиметил)-2-(4-нитроф...	22
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	434A731D-8B98-4800-85A7-4A254E2A8B1D	000023	R-Ондансетрон	23
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	42302830-8E7B-47A9-B073-C7B1EB4AE43C	000024	S)Амлодипин	24
2019-01-01 00:00:00.000	2222-01-01 00:00:00.000	D30202C3-B68E-4F74-8423-03CD42894E16	000025	S-Пантопрозол	25

Рисунок 33 – Таблица oms\_onco\_n020 в БД после проведения протоколов

По результату проведения протокола в таблицу oms\_onco\_n020 добавилось 2471 запись.

Таблица oms\_onco\_n001 в БД после проведения протоколов изображена на рисунке 34.

DATEBEG	DATEEND	GUIDN001	ID_PcOr	onco_N001ID	PcOr_NAME	x_Edition	x_Status
1900-01-01 00:00:00.000	2222-01-01 00:00:00.000	00000000-0000-0000-0000-000000000000	0	0		1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	ED16C80C-7879-450A-96D2-A508CBFCAD5F	1	1	Противопоказания к проведению хирургического лече...	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	F847174A-3A67-4BD3-B177-4F270DAD5A85	2	2	Противопоказания к проведению химиотерапевтическ...	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	DA58E6B4-8C42-4F8A-A325-D7C8BCEE2A58	3	3	Противопоказания к проведению лучевой терапии	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	56969BB5-16FA-4A95-999C-DD4BD2809450	4	4	Отказ от проведения хирургического лечения	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	4A62F915-E4D9-4E40-836F-C6538F22FF45	5	5	Отказ от проведения химиотерапевтического лечения	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	998F98EE-F8AA-4FC0-8A50-7C685485D1F9	6	6	Отказ от проведения лучевой терапии	1	1
1900-01-01 00:00:00.000	2019-01-01 00:00:00.000	086C3308-EDC3-4D16-9885-20787BC5CE17	10	7	тест	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	DA17357D-0885-429E-9612-8C67A1145227	7	94	Гистологическое подтверждение диагноза не показано	1	1
2018-09-01 00:00:00.000	2222-01-01 00:00:00.000	C3EB7EBC-A9D9-4DFE-8331-E41C7A8D547D	8	95	Противопоказания к проведению гистологического ис...	1	1

Рисунок 34 – Таблица oms\_onco\_n001 в БД после проведения протоколов

В итоге можно сделать вывод что ожидаемые и реальные результаты обработки нормативно-справочной информации совпадают.

Результат работы протокола выгрузки реестров счетов представлен на рисунке 35.

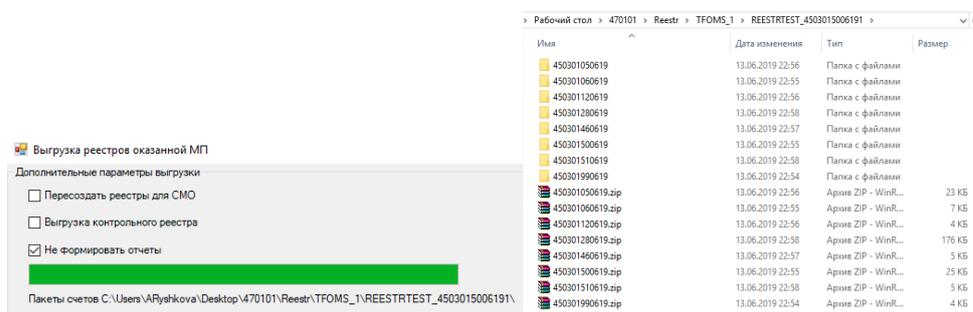


Рисунок 35 – Результат работы протокола выгрузки реестров счетов

В результате мы получили необходимые файлы по всем страховым компаниям для дальнейшей сдачи реестров счетов.

В данном разделе была описана программная реализация и тестирование протоколов обработки нормативно-справочной информации и выгрузки реестров счетов. Приведена и подробно описана структурная схема БД, диаграммы, описывающие системы в целом, приведены результаты тестирования.

## ЗАКЛЮЧЕНИЕ

В заключении следует подвести итоги совершенной работы. Были закреплены теоретические знания, полученные в процессе обучения в университете. Выполнены и подробно описана краткая характеристика работы отдела взаиморасчетов, описана структура работы системы, характеристика выявленных недостатков, постановка задач для устранения выявленных недостатков, информационная модель и ее описание, структура БД, обоснование выбора технологий для разработки, программная реализация проектных решений, общие положения разработки, контрольный пример реализации.

Цель выпускной квалификационной работы – усовершенствование работы отдела взаиморасчетов при выгрузке реестров счетов, за счет поддержания автоматизированной обработки пакета НСИ.

Задачи, решаемые в выпускной квалификационной работе: изучение предметной области; выявление недостатков существующего механизма обработки НСИ и выгрузки реестров счетов обоснование выбора основных проектных решений; обоснование необходимости создания протоколов обработки НСИ и выгрузки реестров счетов; разработка протоколов обработки нормативно-справочной информации и выгрузки реестров счетов для медицинской организации Московской области; тестирование разработанного протокола выгрузки реестров счетов в тестовый период и обработки НСИ.

В данной выпускной квалификационной работе были рассмотрены вопросы: общие сведения о предприятии и его организационная структура; характеристика, задачи и функции, выполняемые подразделением, для которого разрабатывались протоколы; обоснование проектных решений по техническому обеспечению; обоснование проектных решений по информационному обеспечению; обоснование проектных решений по программному обеспечению; обоснование проектных решений по технологическому обеспечению; обоснование выбора программных средств.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кликер, Д.А. Системология. Автоматизация решения системных задач: пер. с англ. / Д.А. Кликер. – Липецк: Книжный бульвар, 2014. – 340с.
2. Рон, Хардман Oracle Database PL/SQL. Рекомендации эксперта / Хардман Рон. - М.: ЛОРИ, 2014. - 10 с
3. Саураб, Гупта Oracle PL/SQL. Руководство для разработчиков / Гупта Саураб. - М.: ЛОРИ, 2014. - 15 с
4. Власов, Норманно-справочная информация: проверено практикой/ М.В.Власов. – Москва: Форум, 2011. – 2.
5. Селко, Джо SQL для профессионалов. Программирование / Джо Селко. - М.: ЛОРИ, 2015. - 464 с.
6. Бегг, К.Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. / К.Т. Бегг. – Екатеринбург: Издательство АРМ, 2016. – 345с.
7. Корнеев, В.В. Базы данных. Интеллектуальная обработка информации. / В.В. Корнеев, А.Ф. Гареев, С.В. Васютин. – Москва: Нолидж, 2015. – 258с.
8. Голицына, О.Л. Базы данных / О.Л. Голицына, Н.В. Максимов, И.И. Попов. – Москва: Форум, 2014. – 289с.
9. Варламов, О.О. Эволюционные базы данных и знаний для адаптивного синтеза интеллектуальных систем. Миварное информационное пространство / О.О. Варламов. – Москва: Радио и связь, 2014. – 341с.
10. Советов, Б.Я. Базы данных: теория и практика. Общество с ограниченной ответственностью / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовской. – Москва: Издательство ЮРАЙТ, 2016. – 439с.
11. Дюбуа, П.С. MySQL: Полное и исчерпывающее руководство по применению и администрированию баз данных MySQL 4, а также программированию приложений. / П.С. Дюбуа. – Москва: Издательский дом Вильямс, 2014. – 532с.

12. Яргер, Р.Д. MySQL и mSQL. Базы данных для небольших предприятий и Интернета. / Р.Д. Яргер, Д.В. Риз, Т.Л. Кинг. – СПб: Символ-Плюс, 2017. – 538с.
13. Миронов, В.В. Ситуационно-ориентированные базы данных: концепция, архитектура, XML-реализация / В.В. Миронов, Н.И. Юсупова, Г.Р. Шакирова. – Уфа: Вестник, 2015. – 376с.
14. Маклаков, С.В. BPwin и ERwin. CASE-средства разработки информационных систем. / С.В. Маклаков. – Москва: Диалог-МИФИ, 2017. – 412с.
15. Организационно-технологический регламент информационного взаимодействия автоматизированных информационных систем участников ОМС Московской области. 2019 [Электронный ресурс]. URL: <http://www.mofoms.ru/documents/OTR/index.php>. (Дата обращения: 20.05.2019).
16. Организационно-технологический регламент информационного взаимодействия автоматизированных информационных систем участников ОМС Московской области. ОТР-ИВ-1.1906 от 05.06.2019 [Электронный ресурс]. URL: [http://www.mofoms.ru/upload/iblock/024/OTR\\_ИВ\\_1\\_1906.pdf](http://www.mofoms.ru/upload/iblock/024/OTR_ИВ_1_1906.pdf). (Дата обращения: 21.05.2019).
17. Руководство пользователя Медицинская информационная система на базе программного комплекса «ТрастМед». – Белгород: Руководство пользователя ООО «СофТраст», 2010.
18. Грекул, В.И. Проектирование информационных систем / В.И. Грекул, Г.Н. Денищенко, Н.Л. Коровкина. – Москва: Интернет-университет информационных технологий-ИНТУИТ, 2015.
19. Якимов, И.М. Комплексный подход к моделированию сложных систем в системе BPwin-Arena / И.М. Якимов. – Казань: Вестник Казанского технологического университета, 2014. – № 6.
20. Маклаков, С.В. Моделирование бизнес-процессов / С.В. Маклаков. – Москва: Диалог, 2016. – 258с.

21. Репин, В.В. Сравнительный анализ нотаций ARIS/IDEF и продуктов, их поддерживающих [Электронный ресурс] / В.В. Репин. – Электрон. текстовые дан. – Москва, 2015. – Режим доступа: Web: [http://www.iteam.ru/publications/it/section\\_51/article\\_2518](http://www.iteam.ru/publications/it/section_51/article_2518), свободный.
22. Атисков, А.Ю. Автоматизированная система трансформации диаграмм бизнес-процессов в диаграммы классов / А.Ю. Атисков. – Самара: Издательства №3, 2016. – 155с.
23. Брайан, Найт Microsoft SQL Server 2005. Руководство администратора / Найт Брайан. - М.: Диалектика, 2008. - 294 с.
24. Браст, Э.Дж. Разработка приложений на основе Microsoft SQL Server 2008 / Э.Дж. Браст. - М.: Русская Редакция, 2010. - 751 с.
25. Вишневский, Алексей Microsoft SQL Server. Эффективная работа / Алексей Вишневский. - М.: Питер, 2009. - 143 с.
26. Долгих, А. Microsoft SQL Server 2005. Практические методы работы (+ CD-ROM) / А. Долгих. - М.: Эком, 2007. - 356 с.
27. Дэвидсон, Луис Проектирование баз данных на SQL Server 2000 / Луис Дэвидсон. - М.: Бинوم. Лаборатория знаний, 2003. - 662 с.
28. Жилинский, А. Самоучитель Microsoft SQL Server 2005 / А. Жилинский. - М.: БХВ-Петербург, 2007. - 224 с.
29. Ицик, Бен-Ган Microsoft SQL Server 2012. Высокопроизводительный код T-SQL. Оконные функции / Бен-Ган Ицик. - М.: Русская Редакция, 2013. - 861 с.
30. Кригель, А. SQL. Библия пользователя / А. Кригель. - М.: Диалектика / Вильямс, 2010. - 326 с.
31. Майк, Хотек Microsoft SQL Server 2008. Реализация и обслуживание. Учебный курс Microsoft (+ CD-ROM) / Хотек Майк. - М.: Русская Редакция, 2012. - 961 с.
32. Морган, С. Проектирование и оптимизация доступа к базам данных Microsoft SQL Server 2005. Учебный курс Microsoft (+ CD-ROM) / С. Морган. - М.: Русская Редакция, 2008. - 550 с.

33. Нильсен, Пол SQL Server 2005. Библия пользователя / Пол Нильсен.  
- Москва: Высшая школа, 2008. - 516 с.
34. Петкович, Д. Microsoft SQL Server 2012. Руководство для начинающих / Д. Петкович. - М.: БХВ-Петербург, 2013. - 816 с.
35. Петкович, Душан Microsoft SQL Server 2012. Руководство для начинающих / Душан Петкович. - М.: БХВ-Петербург, 2012. - 460 с.

## ПРИЛОЖЕНИЕ А

Этап 0.2. Загрузка результатов ФЛК

Отчет о процессе выполнения: Сохранить Распечатать

Загруженные ошибки

Источник	Код ошибки	Описание ошибки	Название полей	Количество ошибок
файл с	D1	Заполнение B_PROT не соответствует регламенту.	B_PROT	138
файл с	D1	Заполнение BSA не соответствует регламенту.	BSA	35
файл с	D1	Заполнение DATE_INJ не соответствует регламенту.	DATE_INJ	34
файл с	D1	Заполнение DIAG_DATE не соответствует регламенту.	DIAG_DATE	84
файл с	D1	Заполнение DT_CONS не соответствует регламенту.	DT_CONS	138
файл с	D1	Заполнение HEI не соответствует регламенту.	HEI	35
файл с	D1	Заполнение K_FR не соответствует регламенту.	K_FR	138
файл с	D1	Заполнение LEK_TIP_L не соответствует регламенту.	LEK_TIP_L	33
файл с	D1	Заполнение NAPR_V не соответствует регламенту.	NAPR_V	138
файл с	D1	Заполнение ONK_M не соответствует регламенту.	ONK_M	23
файл с	D1	Заполнение ONK_N не соответствует регламенту.	ONK_N	23
файл с	D1	Заполнение ONK_T не соответствует регламенту.	ONK_T	23
файл с	D1	Заполнение STAD не соответствует регламенту.	STAD	27
файл с	D1	Заполнение USL_TIP не соответствует регламенту.	USL_TIP	35
файл с	D1	Заполнение WEI не соответствует регламенту.	WEI	35
файл g	S1	Для услуги КСГ или КПГ неверно указано значение DKK1	DKK1	1
файл m	S1	Для суммы «всех» коэффициентов Кдвн мероприятий с ISP_DVN#03 не выполняется условие >= 0.85.	ISP_DVN	36
файл m	S1	Сумма «всех» коэффициентов Кдвн услуг (мероприятий) не равна 1.	OT_PER, CODE_LPU, MSK_OT, PERSCODE, NHISTORY, NEW_DVN	36
файл u	D1	Заполнение поля C_ZAB не соответствует регламенту.	C_ZAB	941
файл u	R1	Тариф отсутствует.	CODE_LPU, CODE_USL, VID_MP, USL_OK, VOZR, PROFIL	20
файл u	S1	Случай оказания стомат. помощи должен содержать хотя бы одну услугу приема	PROFIL	3
файл u	S1	Возрастная категория не соответствует услуге	CODE_USL	13
файл u	S1	Для услуг относящихся к амбулаторным условиям оказания при ED_I2M = 18, не верно указан МКБ1	МКБ1	5
файл u	S1	Для услуги должна быть задана зубная формула	CODE_USL	442
файл u	S1	Значение DS_ONK не соответствует регламенту.	DS_ONK	3
файл u	S1	Неверно рассчитан базовый тариф.	TARIF_B	245
файл u	S1	Неверно указана дата начала оказания медуслуги.	DATE_IN	25
файл u	S1	Суммарное значение количества оказанных медицинских услуг, представленные к оплате, превышают предельно установленные объемы.	KOL_USL	16609
файл u	S1	Услуга дистансеризации не соответствует полу или возрасту.	CODE_USL	77

Рисунок 36 – Результат проверки ФЛК до обновления НСИ

Этап 0.2. Загрузка результатов ФЛК

Отчет о процессе выполнения: Сохранить Распечатать

Загруженные ошибки

Источник	Код ошибки	Описание ошибки	Название полей	Количество ошибок
файл u	S1	Суммарное значение количества оказанных медицинских услуг, представленные к оплате, превышают предельно установленные объемы.	KOL_USL	1980

Рисунок 37 – Результат проверки ФЛК после обновления НСИ

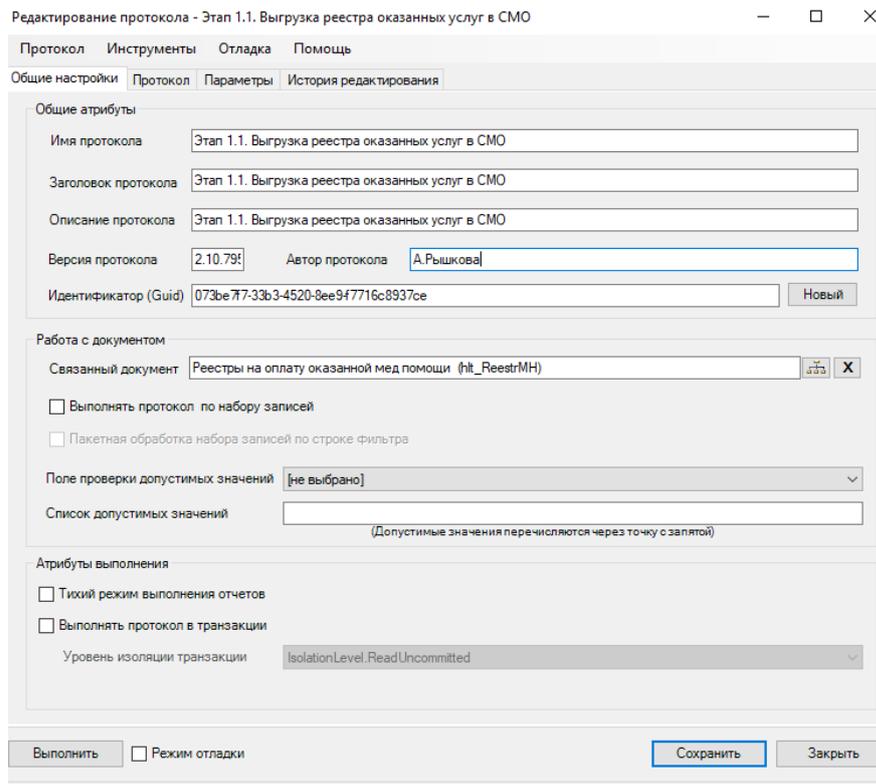


Рисунок 38 – Общие настройки протокола выгрузки реестров счетов

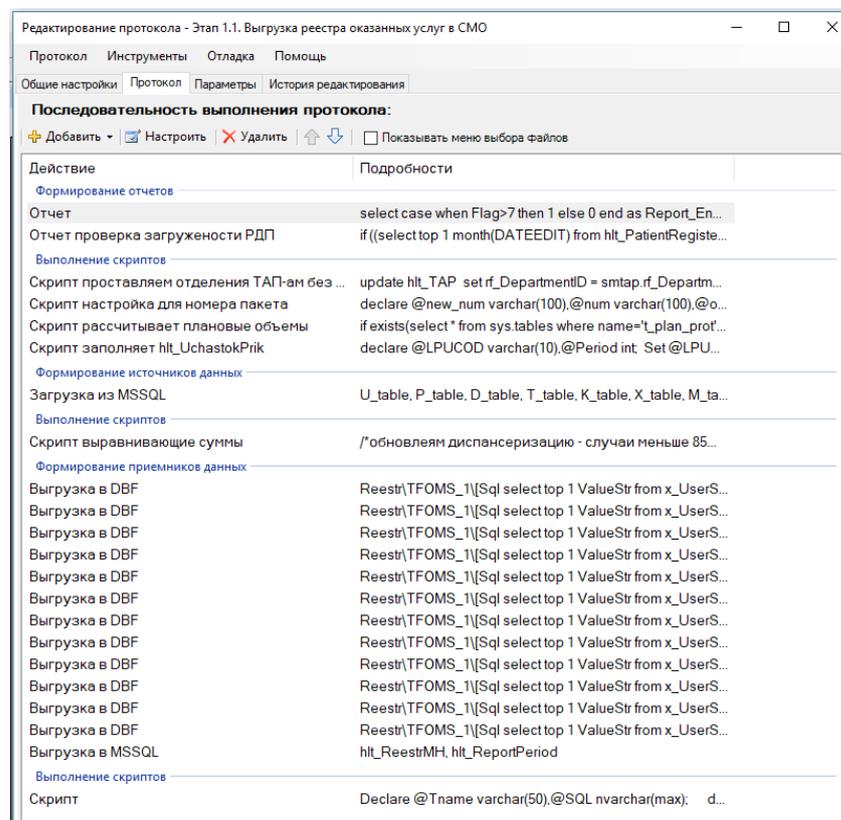


Рисунок 39 – Протокол выгрузки реестров счетов

Дата начала	Дата окончания	Номер счета	Дата счета	Статус	Примечание	СМО	Сумма выставлена	Количество выставлено
01.09.2018	30.09.2018	48	02.10.2018		7 Счета за оказание МП в АО "СТРАХОВАЯ КОМПАНИЯ "СОГАЗ-МЕД"	АО "СТРАХОВАЯ КОМПАНИЯ "СОГАЗ-МЕД"		0
01.09.2018	30.09.2018	48	02.10.2018		7 Счета за оказание МП в ООО ВТБ ИС	ООО ВТБ ИС	2712296	
01.09.2018	30.09.2018	48	02.10.2018		7 Счета за оказание МП в ООО "РГС-МЕДИЦИНА"	ООО "РГС-МЕДИЦИНА"		0
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП по неидентифицированным			87321
01.08.2018	31.08.2018	47	08.09.2018		7 Услуги оказанные интотерриториальным пациентам			837549
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП в АО "МАКС-И"	АО "МАКС-И"		164483
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП в ООО "МСК "МЕДСТРАХ"	ООО "МСК "МЕДСТРАХ"		0
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП в ООО "СМК РЕСО-МЕД"	ООО "СМК РЕСО-МЕД"		238759
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП в АО "МСК "УРАЛСИБ"	АО "МСК "УРАЛСИБ"		48384
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП в АО "СТРАХОВАЯ КОМПАНИЯ "СОГАЗ-МЕД"	АО "СТРАХОВАЯ КОМПАНИЯ "СОГАЗ-МЕД"		0
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП в ООО ВТБ ИС	ООО ВТБ ИС		4377894
01.08.2018	31.08.2018	47	08.09.2018		7 Счета за оказание МП в ООО "РГС-МЕДИЦИНА"	ООО "РГС-МЕДИЦИНА"		0
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП по неидентифицированным			47885
01.07.2018	31.07.2018	46	02.08.2018		4 Услуги оказанные интотерриториальным пациентам			516276
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП в АО "МАКС-И"	АО "МАКС-И"		167106
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП в ООО "МСК "МЕДСТРАХ"	ООО "МСК "МЕДСТРАХ"		0
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП в ООО "СМК РЕСО-МЕД"	ООО "СМК РЕСО-МЕД"		388336
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП в АО "МСК "УРАЛСИБ"	АО "МСК "УРАЛСИБ"		42557
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП в АО "СТРАХОВАЯ КОМПАНИЯ "СОГАЗ-МЕД"	АО "СТРАХОВАЯ КОМПАНИЯ "СОГАЗ-МЕД"		29435
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП в ООО ВТБ ИС	ООО ВТБ ИС		3374361
01.07.2018	31.07.2018	46	02.08.2018		4 Счета за оказание МП в ООО "РГС-МЕДИЦИНА"	ООО "РГС-МЕДИЦИНА"		0
01.06.2018	30.06.2018	45	06.07.2018		4 Счета за оказание МП по неидентифицированным			118305
01.06.2018	30.06.2018	45	06.07.2018		4 Услуги оказанные интотерриториальным пациентам			522409
01.06.2018	30.06.2018	45	06.07.2018		4 Счета за оказание МП в АО "МАКС-И"	АО "МАКС-И"		143870
01.06.2018	30.06.2018	45	06.07.2018		4 Счета за оказание МП в ООО "МСК "МЕДСТРАХ"	ООО "МСК "МЕДСТРАХ"		0
01.06.2018	30.06.2018	45	06.07.2018		4 Счета за оказание МП в ООО "СМК РЕСО-МЕД"	ООО "СМК РЕСО-МЕД"		217196

Рисунок 40 – Запуск протокола выгрузки реестров счетов

Этап 1.1. Выгрузка реестра оказанных услуг в СМО

Подробности процесса выполнения:

Общее время работы: 00:01:19

Процесс запущен в 14:17:14  
Идет предварительная обработка данных...  
Обрабатываются источники данных. Операция может занять продолжительное время.  
Обрабатываются приемники данных. Операция может занять продолжительное время.  
Выгружено 1291 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\Р050619.DBF'  
Выгружено 7155 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\U050619.DBF'  
Выгружено 51 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\D050619.DBF'  
Выгружено 1 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\K050619.DBF'  
Выгружено 0 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\T050619.DBF'  
Выгружено 0 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\X050619.DBF'  
Выгружено 0 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\M050619.DBF'  
Выгружено 0 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\G050619.DBF'  
Выгружено 4226 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\Z050619.DBF'  
Выгружено 0 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\N050619.DBF'  
Выгружено 0 записей в файл 'C:\Users\ARyshkova\Desktop\470101\Reestr\TFOMS\_1\REESTRTEST\_2005015006191\200501050619\C050619.DBF'  
0 записей было обновлено в таблице [Реестры на оплату оказанной мед помощи]  
0 записей было обновлено в таблице [Отчетный период]  
Процесс завершен за 00:01:21

Операция успешно завершена.

Выполнение завершено

Перезапустить    Закрыть

Рисунок 42 – Результат запуска протокола выгрузки реестров счетов

## ПРИЛОЖЕНИЕ В

### Листинг 5 – построение временных таблиц

```
select ROW_NUMBER()over(order by NHISTORY,CODE_USL) as N_Rec,
       t_amb.*,
       convert(varchar(17),'00.0000.00000.000') as TAL_HMP,
       convert(datetime,NULL) as DATE_HMP,
       convert(decimal(1,0),NULL) as INV

from(select
0 as MigrationPatientID,

      0 as MedServicePatientID,

      '@mm' + '@yy' as [OT_PER],

      convert(varchar(6),isnull(left(oms_LPU.MCOD,6),'@FCOD')) as CODE_LPU,

      convert(varchar(2),'@PLAT_CODE') as MSK_OT,

      convert(varchar(15),case when hlt_TAP.rf_polisMKABID=0 then
'M'+convert(varchar(14),hlt_TAP.rf_MKABID)

          else 'A'+convert(varchar(14),hlt_TAP.rf_polisMKABID)
          end) as PERSCODE,

      convert(varchar(2),case when ServiceMedicalCode = '2719600' then /*13*/ '11'
                               /*in ('1719602',
'2719602') then '13'*/
                               when left(ServiceMedicalCode,3) = '2719600'
and ltrim(rtrim(depProf.Code)) = '160' then '13' /*in ('271','171') and
ltrim(rtrim(depProf.Code)) = '72' then '13'*/
                               -- хз зачем          when
left(ServiceMedicalCode,3) in ('271','171') then '11'
                               when depProf.Code in ('68','57','97') then
'12'
                               when left(ServiceMedicalCode,3) in
('109','136','110','209','236','210') then '12'-- !!!!!
                               when left(ServiceMedicalCode,3) = '384' then
'21' else '13' end
      ) as VID_MP,
      /*Вид медицинской помощи* +*/
select *

      convert(varchar(2),'03') as USL_OK,
      /*+*/

      convert(varchar (15),mTAPID) as NHISTORY,

      convert(varchar(3),case when Info='UE' then case
                               when exists ( Select 1 from hlt_SMTAP
                                               inner join
          oms_ServiceMedical on ServiceMedicalID=rf_omsServiceMedicalID and (ServiceMedicalName
like '%зубн%врач%' or ServiceMedicalName like '%гигиенист%')
                                               and
rf_TAPID=mTAPID
                               ) then '85' /*'16'*/
```

```

                                when exists (      Select 1 from hlt_SMTAP
                                                    inner join
oms_ServiceMedical on ServiceMedicalID=rf_omsServiceMedicalID and (ServiceMedicalName like
'%Физиотерап%') and rf_TAPID=mTAPID
                                                    ) then '109' /*'88'*/
                                when prvs_name like '%стомат%дет%'
then '86'          /*СТОМАТОЛОГИЯ ДЕТСКАЯ*/
                                when prvs_name like '%Стомат%общ%практ%' then '171'
                                when prvs_name like '%Физиотерапия%'      then
'109'          /*'88' Физиотерапия*/
                                when prvs_name like '%Челюс%лиц%хир%'      then
'116'          /*'35'Челюстно-лицевая хирургия*/
                                when prvs_name like '%стомат%тер%'
then '89'          /*СТОМАТОЛОГИЯ ТЕРАПЕВТИЧЕСКАЯ*/
                                when prvs_name like '%стомат%хир%'
then '90'          /*СТОМАТОЛОГИЯ ХИРУРГИЧЕСКАЯ*/

                                when prvs_name like '%ОРТОДОН%'
then '63'          /*ОРТОДОНТИЯ*/
                                when prvs_name like '%стомат%профилакт%' then '87'
/*СТОМАТОЛОГИЯ ПРОФИЛАКТИЧЕСКАЯ*/
                                when prvs_name like '%Анестезиология%'      then '5'
/*Анестезиология*/

                                else '85' end
                                else case      when ServiceMedicalCode='B01.027.001'
then '60'
                                when
ServiceMedicalCode = '2719600' then '42'
                                when ServiceMedicalCode =
'2729601' then '160'
                                when
ServiceMedicalName like '%реабилитац%' then '158'
                                /*'68' *Реабилитация*/
                                when
depProf.kl_DepartmentProfileID=0 OR depProf.Code='не определено' then '57' /*'36'ОВП*/
                                else
ltrim(rtrim(depProf.Code)) end
                                /*++*/
                                end) as PROFIL,

                                convert(varchar(6),left(Replace(case when ds_3 is null then mkbosn.DS when kol_mkb>1
then ds_3+'0' else mkbosn.DS end,','),4)) as MKB1,
                                convert(varchar(6),left(Replace(mkbdop.DS,','),4)) as MKB2,
                                convert(varchar(6),'') as MKB3,
                                convert(varchar(15),ltrim(rtrim(ServiceMedicalCode))) as CODE_USL,
                                convert(varchar(6),'') as CODE_MD,
/*Медицинский работник*/
                                hlt_TAP.rf_DocPRVDID as DocPRVDID,

                                case when DATE_IN>DATE_OUT then DATE_OUT else DATE_IN end as DATE_IN,      /*Дата
начала оказания медицинской услуги */

                                case when DATE_IN>DATE_OUT then DATE_IN else DATE_OUT end as DATE_OUT,      /*Дата
окончания оказания медицинской услуги */

                                convert(decimal(6,2),case when Info='UE' and KOL_USL>0 then KOL_USL else 1 end) as
KOL_USL,

```

```

convert(decimal(3,0),case when Info='UE' OR KOL_FACT=1 and '@FCOD' not in
('990201','990501') and mcu.Code<>'09' or info='PP' then 0 else KOL_FACT end) as KOL_FACT,

case when hlt_TAP.IsClosed = '0' or VisitResult.CODE = '0' OR VisitResult.CODE = ''
then '301' else VisitResult.CODE end as ISH_MOV, /* ++ */

convert(varchar(3), case when hlt_TAP.IsClosed = '0' or statCureResult.Code = '0'
OR statCureResult.Code = '' then '303' else statCureResult.Code end ) as RES_GOSP,
/*Результат обращения*/ /* ++ */

convert(varchar(2), case
when '@PLAT_CODE'='99' then '15'
/*сводная справка к реестру счетов медицинской помощи, оказанной пациентам, не
идентифицированным как застрахованные по ОМС лица */
WHEN info='PC' then '00'
when servicemedicalcode = '2309601' then
'09'
when '@PLAT_CODE'='50' and (depProf.Code in
( '73','110','72','74') or (ServiceMedicalCode like '[12]8%' and depProf.Code='16'))
then '17'/*Сводная справка к реестру счетов по сверхбазовой программе ОМС (МТР)
('74','79','80','82','85')*/ /* ++ */
when '@PLAT_CODE'='50' then '12'
/*Счет-фактура "Б"-2009, Сводная справка к Реестру счетов
-"иногородние" с 05.2011 */
when depProf.Code in ( '73','110','72','74')
or (ServiceMedicalCode like '[12]8%' and depProf.Code='16') then '16'/*Сводная справка к
реестру счетов по сверхбазовой программе ОМС *'74','79', 80,'82','85'*/ /* ++ */
when depProf.Code in ('97',
'68','53','20','112', '57','42') and (servicemedicalcode like 'B0[14].0[123][0316].00[1-4]'
or servicemedicalcode like 'B0[14].0[45]7.00[1-6]'
or
servicemedicalcode='2719600') then '14' /*ОТП НОВОЕ*/
when isnull((
Summ_APP from hlt_UchastokPRIK
oms_SMO on SMOID=rf_SMOID
rf_ReportPeriodID=@Document.rf_ReportPeriodID and SMOID=@Document.rf_SMOID and SMOID>0 and
Summ_APP>0 ),0
)=0 then '09'
/*не имеющие прикрепленного населения */

when ServiceMedicalName like '%МЕД%ОСМОТР%'
OR info='DP'
/*ServiceMedicalName like '%диспансеризац%' ++ */
OR info='UE'
/*ServiceMedicalCode like '%D'--стоматология++ */
OR depProf.Code in ('16','30','18','60','160')
/*ДЕРМАТОЛОГИЯ,ОНКОЛОГИЯ,колопроктология,НЕОТЛОЖНАЯ ПОМОЩЬ++ * */
OR servicemedicalcode like '[21]32964[02]%'
/* (ПРЕНАТАЛЬНЫЙ СКРИНИНГ)++ */
OR ServiceMedicalCode in ('2329680', '1329680',
/*ПРОВЕДЕНИЕ ВНУТРИУТРОБНОЙ ЭХОДОППЛЕРКАРДИОГРАФИИ++*/

```

```

'2019604') /*консультация «мобильного» кардиолога»+*/
'1019604',

OR info in
('GD','PD','CD','RM','KT','MT','KT1','MT1','PT','CZ','NP','CDO','CDR') /*ServiceMedicalCode
like '370318%'--диализ таб 11 ОТП7+RM+*/
then '09' /*Счет-фактура "A2"-2009, "A1"
с 2011, Сводная справка к Реестру счетов СМ01 с 05.2011*/
else '09' /*отчет по списанию средств
("подушевое" финансирование, включая ОВП, пилотный проект) с 08.2013*/
end) as VID_SF,

convert(decimal(10,2),isnull((Select top 1 Value1 from oms_Tariff
where Value1>0 and
oms_Tariff.rf_ServiceMedicalID=oms_ServiceMedical.ServiceMedicalID and
Date_E>getdate()),TARIF_B_old)) as TARIF_B,

convert(decimal(10,2),isnull((Select top 1 Value1 from oms_Tariff
where Value1>0 and
oms_Tariff.rf_ServiceMedicalID=oms_ServiceMedical.ServiceMedicalID and
Date_E>getdate()),TARIF_B_old)) as TARIF_S,

convert(decimal(10,2),0) as TARIF_D,

convert(decimal(10,2),isnull((Select top 1 Value1 from oms_Tariff
where Value1>0 and
oms_Tariff.rf_ServiceMedicalID=oms_ServiceMedical.ServiceMedicalID and
Date_E>getdate()),TARIF_B_old)*case when Info='UE' and KOL_USL>0 then KOL_USL else 1.00 end)
as SUM_RUB,

convert(varchar(2),case when tt.CODE='0' then '' else right('0'+cast(tt.CODE as
varchar),2) end) as VID_TR,

convert(decimal(1,0),0) as EXTR,

/*convert(varchar(12),'') as CODE_OTD,*/

convert(varchar(12),DepartmentCode) as CODE_OTD,

convert(decimal(1,0),1) as SOUF,

isnull(MED_AREA,'') as MED_AREA,

isnull(MED_AREA_N,'') as MED_AREA_N,

case when oms_ServiceMedical.ServiceMedicalName like '%Неотлож%' then '2' else '3'
end as For_Pom,

case when nwd.NotWorkDocID is null then 0 else 1 end as MSE,

case
when (typ=0 and KOL_FACT>1) or (st = 1 and KOL_FACT>1 )then '3.0'--обращение
по заболеванию
when len(oms_k1_ReasonType.Code) = 3 then oms_k1_ReasonType.Code
when oms_k1_ReasonType.Code = '3' then '2.5'
when left(Replace(case when ds_3 is null then mkbosn.DS when kol_mkb>1 then
ds_3+'0' else mkbosn.DS end,',''),4) like 'Z%' then '2.1' --when typ>0
then '3.1'--посещение по профилактике
when typ=0 and KOL_FACT>1 then '3.0'--обращение по заболеванию

```

```

        when typ=0 and KOL_FACT in (0,1) then '1.0'--посещение по заболеванию when
left(Replace(case when ds_3 is null then mkbosn.DS when kol_mkb>1 then ds_3+'0' else
mkbosn.DS end, '.', ''),4) not like 'Z%' then '1.0' /* '3.0' */
        when ServiceMedicalName like '%медицинский%осмотр%' then '2.1'
        when oms_ServiceMedical.ServiceMedicalName like '%Неотлож%' then '1.1'
        when oms_ServiceMedical.ServiceMedicalName like '%ДИСПАНСЕРИЗАЦ%' then '2.2'
        else '2.6' end
    as P_CEL,

    case when oms_kl_ReasonType.Code='1.3' then case when oms_kl_DisRegState.Code='0'
then 1
        when oms_kl_DisRegState.Code='3'
then 6 else oms_kl_DisRegState.Code
        end else 0
end as DN,

    case when ServiceMedicalName like '%реабилитац%' then 1 else 0 end as REAB ,

    oms_PRVS.I_PRVS as PRVS,

    convert(varchar(6),NULL) as MCODE_OUT,

    convert(varchar(20),NULL) as NOM_NPR,

    convert(datetime,NULL) as DATE_NPR,

case when
    convert(varchar(5),left(Replace(
        case when ds_3 is null then mkbosn.DS
        when kol_mkb>1 then ds_3+'0'
        else mkbosn.DS end
    , '.', ''),4)) like 'C%' and mcu.code not in ('12', '19') then convert(decimal
(1),2)
    else convert(decimal (1),0) end as DS_ONK,

    case when (mcu.code='11'or left(mkbosn.DS ,1)='Z') then 0
        when hlt_TAP.rf_kl_DiseaseTypeID=0 then 1 else dt1.code end as C_ZAB,

    0 as PROFIL_K,

    case when VisitResult.code='309' /*'38'*/ then
isnull(convert(varchar(6),isnull(left(lpu.MCODE,6),lpu.FCODE)), '') else '' end as NAPR_MO,

        case when month(DATE_IN)< 10
        then '0'+ convert(varchar(2),month(DATE_IN))
        else convert(varchar(2),month(DATE_IN)) end + right(
year(DATE_IN), 2) as OT_PER_U,

        convert(varchar(11),
'A'+convert(varchar(14),hlt_TAP.Tapid)) as IDCASE
from(
Select case when ServiceMedicalCode like '[12][0-9][0-9]960[0123]' and ServiceMedicalCode
not like '[12]7[12]96%' /*неотложка, ФАП*/
and ServiceMedicalCode not like '[12]6796'
/*ПРОВЕДЕНИЕ КОМПЬЮТЕРНОЙ ТОМОГРАФИИ - исключение*/
or ( DS like 'Z%' and ServiceMedicalCode like 'B%' and info<>'UE') /*чтобы не менять у
стоматок*/
        then 0 else 1 end as st, /*иначе не меняем услугу*/

    hlt_MKAB.family,hlt_MKAB.Name,hlt_MKAB.ot,hlt_MKAB.Date_BD,ltrim(rtrim(hlt_MKAB.S_POL
+' '+hlt_MKAB.N_POL)) as polis,max(TAPID) as mTAPID,

```

```

convert(date,min(case when ServiceMedicalName like '%диспансеризац%' then
isnull(d_disp,hlt_TAP.DateClose) else Date_P end)) as DATE_IN,
convert(date,max(case when ServiceMedicalName like '%диспансеризац%' then
hlt_TAP.DateClose else Date_P end)) as DATE_OUT,
sum(hlt_SMTAP.[Count]) as KOL_USL,count(distinct DATE_P) as KOL_FACT,

case when ServiceMedicalCode like '[12][0-9][0-9]960[0123]' and
ServiceMedicalCode not like '[12]7[12]96%/*неотложка, ФАП*/
and ServiceMedicalCode not like
'[12]6796%/*ПРОВЕДЕНИЕ КОМПЬЮТЕРНОЙ ТОМОГРАФИИ - исключение*/
then substring(oms_ServiceMedical.ServiceMedicalCode,1,5) else
oms_ServiceMedical.ServiceMedicalCode end as Serv,

max(isnull(oms_Tariff.Value1,hlt_ReestrMHSMTAP.SUM_V)) as TARIF_B_old,
case when DS like 'Z%' OR ServiceMedicalCode like '[12]7[12]96%' OR
ServiceMedicalCode like '[12]6796%'
OR ServiceMedicalCode not like '[12][0-9][0-9]960[0123]' then TAPID else 0 end
as typ
--,case when oms_ServiceMedical.Info='UE' then hlt_SMTAP.Description else null
end as Teeth
--,isnull(left(convert(varchar(20),Tooth),2),'') as
TEETH_CODE,isnull(right(convert(varchar(20),OCCLUSION),1),'') as OCCLUSION
, fcode_usl, ServiceMedicalCode AS SMCode

from hlt_ReestrMHSMTAP WITH(NOLOCK)
inner join hlt_ReestrTAPMH WITH(NOLOCK) on rf_ReestrTAPMHID = ReestrTAPMHID
inner join hlt_SMTAP WITH(NOLOCK) on rF_SMTAPID = SMTAPID
--left join oms_kl_Teeth WITH(NOLOCK) on kl_TeethID=rf_kl_TeethID and
kl_TeethID>0
inner join hlt_TAP WITH(NOLOCK) on hlt_TAP.TAPID = hlt_SMTAP.rf_TAPID
inner join hlt_MKAB WITH(NOLOCK) on MKABID=hlt_TAP.rf_MKABID
--inner join hlt_DocPRVD WITH(NOLOCK) on hlt_TAP.rf_DocPRVDID=DocPRVDID
inner join oms_MKB WITH(NOLOCK) on MKBID=hlt_TAP.rf_MKBID
inner join oms_ServiceMedical WITH(NOLOCK) on
ServiceMedicalID=hlt_SMTAP.rf_omsServiceMedicalID and oms_ServiceMedical.date_e >
hlt_TAP.dateTap
inner join oms_kl_MedCareUnit mcu WITH(NOLOCK) on kl_MedCareUnitID =
oms_serviceMedical.rf_kl_MedCareUnitID
left join oms_Tariff WITH(NOLOCK) on hlt_SMTAP.rf_TariffID = TariffID
left join (
Select rf_TAPID,min(Date_P) as d_disp from hlt_ReestrMHSMTAP WITH(NOLOCK)
inner join hlt_SMTAP WITH(NOLOCK) on rF_SMTAPID = SMTAPID
inner join oms_ServiceMedical WITH(NOLOCK) on
ServiceMedicalID=hlt_SMTAP.rf_omsServiceMedicalID
where ServiceMedicalCode like '210[68]00001' and
hlt_ReestrMHSMTAP.rf_ReportPeriodID=@Document.rf_ReportPeriodID
group by rf_TAPID
)t_disp on t_disp.rf_TAPID=hlt_SMTAP.rf_TAPID
where oms_ServiceMedical.Info <> 'PC' and rf_MKABID>0 and ServiceMedicalCode
not like '210[68]000%'--ПЕРВЫЙ ЭТАП ДИСПАНСЕРИЗАЦИИ ВЗРОСЛОГО НАСЕЛЕНИЯ
and hlt_ReestrTAPMH.rf_ReportPeriodID=@Document.rf_ReportPeriodID
and hlt_ReestrTAPMH.rf_ReestrMHID=@Document.ReestrMHID

group by case when ServiceMedicalCode like '[12][0-9][0-9]960[0123]' and ServiceMedicalCode
not like '[12]7[12]96%/*неотложка, ФАП*/
and ServiceMedicalCode not like '[12]6796%/*ПРОВЕДЕНИЕ КОМПЬЮТЕРНОЙ ТОМОГРАФИИ -
исключение*/
or ( DS like 'Z%' and ServiceMedicalCode like 'B%' and info<>'UE') /*чтобы не менять
у стоматок*/
then 0 else 1
end,hlt_MKAB.family,hlt_MKAB.Name,hlt_MKAB.ot,hlt_MKAB.Date_BD,ltrim(rtrim(hlt_MKAB.S_POL+'
+hlt_MKAB.N_POL)),

```

```

        case when ServiceMedicalCode like '[12][0-9][0-9]960[0123]' and
ServiceMedicalCode not like '[12]7[12]96%'/*неотложка, ФАП*/
and ServiceMedicalCode not like '[12]6796%'/*ПРОВЕДЕНИЕ КОМПЬЮТЕРНОЙ
ТОМОГРАФИИ - исключение*/
then substring(oms_ServiceMedical.ServiceMedicalCode,1,5) else
oms_ServiceMedical.ServiceMedicalCode end,
        case when DS like 'Z%' OR ServiceMedicalCode like '[12]7[12]96%' OR
ServiceMedicalCode like '[12]6796%'
OR ServiceMedicalCode not like '[12][0-9][0-9]960[0123]' then TAPID else 0
end,
        case when DS like 'Z%' OR ServiceMedicalCode like '[12]7[12]96%' OR
ServiceMedicalCode like '[12]6796%'
OR ServiceMedicalCode not like '[12][0-9][0-9]960[01234]' then cast(TAPID as
varchar)+convert(varchar,/*date_p*/ fcode_usl,104) else '0' end , fcode_usl,
ServiceMedicalCode
/* добавила [01234]' */
--,case when oms_ServiceMedical.Info='UE' then hlt_SMTAP.Description else null end
--
,isnull(left(convert(varchar(20),Tooth),2),'),isnull(right(convert(varchar(20),OCCLUSION),1
),'))
)t
inner join hlt_TAP on TAPID=mTAPID
inner join oms_kl_DiseaseType dt1 ON dt1.kl_DiseaseTypeID=hlt_TAP.rf_kl_DiseaseTypeID
inner join oms_kl_VisitResult VisitResult on VisitResult.kl_VisitResultID =
hlt_tap.rf_kl_VisitResultID

outer apply ( select top 1 dir.rf_lpuid from hlt_Direction dir
where dir.rf_tapid=hlt_TAP.tapid
order by dir.Directionid desc) oa
left join oms_lpu lpu on lpu.lpuid=oa.rf_lpuid
left join hlt_NotWorkDoc nwd WITH(NOLOCK) on hlt_TAP.TAPID=nwd.rf_TAPID and
nwd.DateDirectMSE>'19000101' and nwd.NotWorkDocID>0
inner join oms_kl_DisRegState WITH(NOLOCK) on
hlt_TAP.rf_kl_DisRegStateId=kl_DisRegStateID
inner join oms_Department on DepartmentID=rf_DepartmentID
inner join hlt_DocPRVD on DocPRVDID=hlt_Tap.rf_DOCPRVDID
inner join oms_PRVS on PRVSID=hlt_DocPRVD.rf_PRVSID
left join (
Select rf_LPUID from oms_tehatrvalue
inner join oms_tehatr t2 on t2.tehatrID=oms_tehatrvalue.rf_tehatrID and
t2.Code='0025'
where oms_tehatrvalue.Value='1' and tehatrvalueID in (
Select max(tehatrvalueID) from oms_tehatrvalue
where oms_tehatrvalue.date_E>getdate()
group by rf_LPUID,rf_tehatrID
)
)tehatr on tehatr.rf_LPUID=oms_Department.rf_LPUID
left join oms_LPU on oms_LPU.LPUID=tehatr.rf_LPUID and oms_LPU.LPUID>0
inner join oms_MKB mkbosn WITH(NOLOCK) on hlt_TAP.rf_MKBID = mkbosn.MKBID
inner join oms_kl_ReasonType on kl_ReasonTypeID=hlt_TAP.rf_kl_ReasonTypeID

inner join oms_ServiceMedical on oms_ServiceMedical.ServiceMedicalCode=case
when st=1 and kol_fact<=1 then Serv else
case
when typ > 0 and KOL_FACT>1 and t.SMCode like 'B%' then t.fcode_usl
when typ=0 and KOL_FACT=1 then case when t.fcode_usl ='0' then serv +
'02' else t.fcode_usl end --посещение по заболеванию +++
when typ>0 and DS like 'Z02.[035789]' then case when t.SMCode like
'B%' then 'B04.'+SUBSTRING (t.SMCode, 5,6)+'2'

else Serv+ '03'
end
end

```

```

                when typ>0 and DS like 'Z%' and t.FCode_Us1 like 'B%' then
'B04.'+SUBSTRING (t.FCode_Us1, 5,6)+'2'
                when typ>0 and DS like 'Z%' and t.SMCode like 'B%' then      'B04.'+SUBSTRING
(t.SMCode, 5,6)+'2'
                else
                Serv+
                case when typ=0 and KOL_FACT>1 then '01' --обращение по заболеванию
                when typ>0 then '00'
                посещение по профилактике
                else '00'
                end
            end
        end --на откуп статистам

--inner join oms_ServiceMedical on --ServiceMedicalID=t.rf_omsServiceMedicalID
--oms_ServiceMedical.ServiceMedicalCode=case /*when Serv like '101730[235]D' and DS like
'Z%' then '1017301D'--ПРОФИЛАКТИЧЕСКИЙ ПРИЕМ
--when Serv like '201230[235]D' and DS like 'Z%' then '2012301D'--ПРОФИЛАКТИЧЕСКИЙ ПРИЕМ*/
--when st=1 then Serv else Serv+
--case when typ>0 /*and oms_kl_ReasonType.Code='4'*/ and DS like 'Z02.[035789]' then '03'
--when typ>0 then '00'--посещение по профилактике
--when typ=0 and KOL_FACT>1 then '01'--обращение по заболеванию
--when typ=0 and KOL_FACT=1 /*and (oms_kl_ReasonType.Code='1' OR rf_kl_ReasonTypeID=0)*/
then '02'--посещение по заболеванию
--else '00' end end --на откуп статистам

/*Тип отделения и профиль берем из услуги, если в услуге данные не заполнены, то из
отделения*/
inner join oms_kl_DepartmentProfile depProf WITH(NOLOCK) on depProf.kl_DepartmentProfileID=
oms_ServiceMedical.rf_kl_DepartmentProfileID
/*Вид мед.помощи обязателен к заполнению*/
inner join oms_kl_MedCareType mct WITH(NOLOCK) on kl_MedCareTypeID =
oms_serviceMedical.rf_kl_MedCareTypeID
inner join oms_kl_MedCareUnit mcu WITH(NOLOCK) on kl_MedCareUnitID =
oms_serviceMedical.rf_kl_MedCareUnitID
inner join oms_kl_TraumaType tt WITH(NOLOCK) on hlt_TAP.rf_kl_TraumaTypeID =
tt.kl_TraumaTypeID
inner join oms_kl_StatCureResult statCureResult on hlt_TAP.rf_kl_StatCureResultID =
kl_StatCureResultID

left JOIn (
Select left(DS,3) as ds_3,count(*) as kol_mkb from oms_MKB
where Date_E>getdate()
group by left(DS,3)
)t_mkb on mkbosn.DS=ds_3
inner join oms_MKB mkb dop WITH(NOLOCK) on hlt_TAP.rf_MKB2ID = mkb.dop.MKBBID
/*Участки по ОТП 7.58*/
left join (
Select rf_MKABID,hlt_UChastok.rf_LPUIID,convert(varchar(15),hlt_UChastok.CODE) as
MED_AREA,convert(varchar(100),hlt_UChastok.UchastoCaption) as MED_AREA_N from
hlt_UChastokMKAB
inner join hlt_UChastok on UChastokID=rf_UChastokID
where isDetach=0 and DateOUT>getdate()
)uch on ltrim(rtrim(depProf.Code)) in /*('09','10')*/ ('68','97') and
uch.rf_MKABID=hlt_TAP.rf_MKABID
and oms_Department.rf_LPUIID=uch.rf_LPUIID

...

```

```
)t_amb
```

```
where Code_USL not like '[12]1700%D' /*in ('117000D','217000D' --заказ от королевской  
стоматке 19.10*/
```

```
@ParNeIdent
```

```
/*and (USL_OK=case when '@PLAT_CODE'='99' then '01' else USL_OK end OR
```

```
USL_OK=case when '@PLAT_CODE'='99' then '07' else USL_OK end OR
```

```
USL_OK=case when '@PLAT_CODE'='99' then '03' else USL_OK end and Profil=case when  
'@PLAT_CODE'='99' then '72' else Profil end)*/
```

## Листинг 6 – формирование С таблицы

```
select
```

```
    ROW_NUMBER()over(order by NHISTORY) as N_Rec,  
    u.ot_per,  
    u.CODE_LPU,  
    u.MSK_OT,  
    u.PERSCODE,  
    u.NHISTORY,  
    u.PROFIL,  
    u.CODE_USL,  
    u.DATE_IN,  
    u.DATE_OUT,  
    case when (isnull (n18.ID_REAS, 0)=9 or (usl_ok='03' and isnull (n18.ID_REAS,  
0)=0)) then /*9*/ 5 else isnull(n18.ID_REAS, 0) end as DS1_T,  
    isnull ((CASE WHEN isnull (n18.ID_REAS, 0) <>9 and isnull (n18.ID_REAS, 0) in  
(0,1,2) THEN onco_2.ID_ST ELSE 0 END), 0) as STAD,  
    /* добавлено and [dbo].FullYearAge(p.BIRTHDAY, u.DATE_IN)>17 */  
    isnull ((CASE WHEN isnull (n18.ID_REAS, 0) <>9 and [dbo].FullYearAge(p.BIRTHDAY,  
u.DATE_IN)>17 THEN onco_3.ID_T ELSE 0 END), 0) as ONK_T,  
    isnull ((CASE WHEN isnull (n18.ID_REAS, 0) <>9 and [dbo].FullYearAge(p.BIRTHDAY,  
u.DATE_IN)>17 THEN onco_4.ID_N ELSE 0 END), 0) as ONK_N,  
    isnull ((CASE WHEN isnull (n18.ID_REAS, 0) <>9 and [dbo].FullYearAge(p.BIRTHDAY,  
u.DATE_IN)>17 THEN onco_5.ID_M ELSE 0 END), 0) as ONK_M,  
    isnull (CASE WHEN isnull (n18.ID_REAS, 0) in (1,2) THEN isnull (tal.mtstz, 0) END,  
0) as MTSTZ,  
    isnull (diag_tip.diag_tip, 0) as DIAG_TIP,  
    isnull (diag_tip.DIAG_CODE, 0) as DIAG_CODE,  
    isnull (diag_tip.DIAG_RSLT, 0) as DIAG_RSLT,  
  
    CASE WHEN ((isnull (onco_1.ID_PROT, 0)<>0) and (isnull (n18.ID_REAS, 0) <>9)) THEN  
onco_1.ID_PROT ELSE 0 END as PROT,  
    /* OTP 7.72 isnull (onco_1.ID_PROT, 0) as PROT, */  
    isnull (po.date, null) as D_PROT,  
    null as NAPR_DATE,  
    0 as NAPR_V,  
    0 as MET_ISSL,  
    '0' as NAPR_USL,  
    isnull (/*CASE WHEN isnull(onco_1.ID_PROT, 0)=0 THEN */onco_19.ID_CONS /*ELSE 0  
END*/, 0) as PR_CONS, /* хз откуда, в отр этого нет */  
    CASE when u.usl_ok='03' and isnull (n18.ID_REAS, 0)<>9 then 6  
        WHEN isnull(n18.ID_REAS, 0) <>9 THEN isnull (s.USL_TIP,0) else  
0 end as USL_TIP,  
    /* OTP 7.72 isnull (s.USL_TIP,0) as USL_TIP, */  
    isnull (CASE WHEN isnull(s.USL_TIP,0)=1 THEN isnull(onco_14.ID_THir, 0) else 0  
end, 0) as HIR_TIP,  
    isnull (CASE WHEN isnull(s.USL_TIP,0)=2 THEN isnull (onco_15.ID_TLek_L, 0) else 0  
end, 0) as LEK_TIP_L,  
    isnull (CASE WHEN isnull(s.USL_TIP,0)=2 THEN isnull (onco_16.ID_TLek_V, 0) else 0  
end, 0) as LEK_TIP_V,
```

```

        isnull (CASE WHEN isnull(s.USL_TIP,0) in (3,4) THEN isnull (onco_17.ID_TLuch, 0)
else 0 end, 0) as LUCH_TIP,
        convert (numeric(6,2),CASE WHEN s.USL_TIP in (3,4) THEN sTal.Sod ELSE 0 END) as SOD,
        CASE WHEN u.p_cel<>'1.3' and tal.Revealed=1 then 0 else 1 end as DS1_F,
        u.date_in as DIAG_DATE,
        isnull(diag_tip.REC_RSLT, 0) as REC_RSLT,
        case when tc.date between '20000101' and '21000101' then tc.date else null end as
DT_CONS,
        isnull (CASE WHEN isnull(s.USL_TIP,0)=2 then isnull(right('000',6-
len(cast(isnull(onco_20.ID_LEKP,0) as varchar)))+cast(isnull(onco_20.ID_LEKP,0) as varchar)
/*onco_20.ID_LEKP*/, '' ) else '' end, '' )as REGNUM,
        isnull (CASE WHEN isnull(s.USL_TIP,0)=2 then isnull(talNS.Date_B, null) else null
end, null ) as DATE_INJ,
        CASE WHEN isnull (onco_1.ID_PROT, 0)<>0 THEN 1 ELSE 0 END as B_PROT,
        null as K_FR,
        CASE WHEN isnull(s.USL_TIP,0)=2 then (sTal.WEI ) else null end as WEI,
        CASE WHEN isnull(s.USL_TIP,0)=2 then (sTal.HEI ) else null end as HEI,
        CASE WHEN isnull(s.USL_TIP,0)=2 then (sTal.BSA) else null end as BSA,
        isnull(sTal.PPTR,0) as PPTR,
case when

CASE WHEN isnull(n18.ID_REAS, 0) <>9 THEN isnull (s.USL_TIP,0) else 0 end in (2,4) and
(u.CODE_USL like 'st05.00[679]' or u.CODE_USL like 'st05.01[01]' or
u.CODE_USL like 'ds05.00[34678]'
or [dbo].FullYearAge(p.BIRTHDAY, u.DATE_IN)<17) then 'нет' else
CodeKSGCriterion end as CODE_SH,

        isnull( case when
                CASE WHEN isnull(n18.ID_REAS, 0) <>9
                        THEN
                                CASE when u.usl_ok='03'
                                        then 6
                                        else isnull (s.USL_TIP,0)
                                end
                        else 0
                end in (1,3,4,6)
                then nServ.CODE
                else ''
        end, '' ) as VID_VME ,

        null as NUM_CONS

from u_table u
        inner join p_table p ON p.PERSCODE=u.perscode
        /*НОВОЕ 04.04.2019*/
        inner join oms_ServiceMedical sm on sm.ServiceMedicalCode = u.CODE_USL and sm.Date_E
>u.DATE_IN
        left join oms_kl_NomService nServ on nServ.kl_NomServiceID =
sm.rf_kl_NomServiceID and nServ.kl_NomServiceID>0

        left join hlt_tap tap on convert(varchar,tap.tapid)=u.NHISTORY
        left join stt_MigrationPatient MigP ON MigP.migrationpatientid =
u.migrationpatientid and MigP.migrationpatientid>0
        left join stt_MedicalHistory mh ON mh.MedicalHistoryID =
MigP.rf_MedicalHistoryID
        left join onco_Talon tal ON case when (tal.rf_MedicalHistoryID=0 or
tal.rf_MedicalHistoryID is null) then tal.rf_tapid else tal.rf_MedicalHistoryID end =
isnull(mh.MedicalHistoryID, tap.tapid)

        left join onco_SmTalon sTal ON sTal.RF_TALONID=TAL.TALONID
        left join onco_smTalonLS talNS ON
talNS.rf_smTALONID=sTal.smTALONID and talNS.rf_onco_N020ID>0
        left join oms_onco_N020 onco_20 ON
onco_20.onco_N020ID=talNS.rf_onco_N020ID

```

```

left join oms_ExpertKSGCriterion KSGcrit on
MigP.rf_ExpertKSGCriterionID = KSGcrit.ExpertKSGCriterionID
--
left join oms_onco_N021 onco_21 ON
onco_21.rf_onco_N020ID=onco_20.onco_N020ID
left join stt_MedservicePatient MedP ON
MedP.MedservicePatientid = sTal.rf_MedservicePatientid
left join oms_onco_N002 onco_2 ON onco_2.onco_N002ID =
tal.rf_onco_N002ID
/* поднимаем по ссылке стадии заболевания
для поля STAD */
left join oms_onco_N005 onco_5 ON onco_5.onco_N005ID =
tal.rf_onco_N005ID
/* поднимаем по ссылке значение Metastasis
для поля ONK_M */
left join oms_onco_N004 onco_4 ON onco_4.onco_N004ID =
tal.rf_onco_N004ID
/* поднимаем по ссылке значение Nodus для
поля ONK_N */
left join oms_onco_N003 onco_3 ON onco_3.onco_N003ID =
tal.rf_onco_N003ID
/* поднимаем по ссылке значение Тumor для
поля ONK_T */
left join (
select onco_18.ID_REAS, talonid from
onco_Talon t
left join oms_onco_N018 onco_18
ON onco_18.onco_N018ID = t.rf_onco_N018ID /* поднимаем по ссылке повод обращения для поля
DS1_T */
) n18 on n18.talonid=tal.talonid
left join (
select
talonid,
CASE WHEN onco_7.ID_mrf=0 THEN 0 ELSE
CASE WHEN onco_8.ID_R_M=0 THEN 0 ELSE
/* если результатов
CASE WHEN onco_8.ID_R_M=0 THEN 0 ELSE
/*
1 END as diag_tip,
если результатов гистологических исследований нет, то 0 */
/* CASE WHEN onco_8.ID_R_M=0 THEN
convert(date,mrf.Diag_Date_Mrf,104)
ELSE null END as
DIAG_DATE, новое из OTP 7.73 */
CASE WHEN onco_7.ID_mrf=0 THEN 0 ELSE
1 END as REC_RSLT
from onco_Talon t
inner join onco_Mrf mrf ON
mrf.rf_talonid=t.talonid
inner join oms_onco_N007 onco_7
ON onco_7.onco_N007ID=mrf.rf_onco_N007ID /* поднимаем по ссылке гистологические
признаки для поля DIAG_CODE */
inner join oms_onco_N008 onco_8
ON onco_8.onco_N008ID=mrf.rf_onco_N008ID /* поднимаем по ссылке результатов
гистологических исследований для поля DIAG_RSLT */
union
select talonid,
CASE WHEN onco_10.ID_igh=0 THEN 0 ELSE
CASE WHEN onco_11.ID_R_I=0 THEN 0 ELSE
/* если результатов значений маркёров
CASE WHEN onco_11.ID_R_I=0 THEN 0 ELSE
/* если
2 END as diag_tip,
результатов значений маркёров нет, то 0 */

```

```

convert(date, lgh.Diag_Date_Lgh, 104)
/*CASE WHEN onco_11.ID_R_I=0 THEN
ELSE null END as DIAG_DATE, новое из OTP 7.73 */
CASE WHEN onco_10.ID_igh=0 THEN 0 ELSE
1 END as REC_RSLT

from onco_Talon t
inner join onco_Lgh lgh on

lgh.rf_TalonId=t.talonid
inner join oms_onco_N010 onco_10 ON
/* поднимаем по ссылке маркёров для поля
onco_10.onco_N010ID=lgh.rf_onco_N010ID
DIAG_CODE */

inner join oms_onco_N011 onco_11 ON
/* поднимаем по ссылке значений маркёров для
onco_11.onco_N011ID=lgh.rf_onco_N011ID
поля DIAG_RSLT */

) diag_tip ON diag_tip.talonid = tal.talonid

left join onco_PrOt po ON po.rf_talonid = tal.talonid

left join onco_TalonCons tc ON tc.rf_talonid = tal.talonid
left join oms_onco_N001 onco_1 ON
onco_1.onco_N001ID = po.rf_onco_N001ID /* поднимаем по ссылке
классификатор противопоказаний и отказов для поля PROT */
left join oms_onco_N019 onco_19 ON
onco_19.onco_N019ID = tc.rf_onco_N019ID /* поднимаем по ссылке
классификатор целей консилиума для поля PR_CONS */

outer apply
(
THEN onco_13.ID_TLech ELSE 0 END, 0) as USL_TIP, st.SmTalonID
from onco_SmTalon st
left join oms_onco_N013 onco_13 ON
onco_13.onco_N013ID = sT.rf_onco_N013ID /* поднимаем по ссылке классификатор
типов лечения для поля USL_TIP */

where st.SmTalonID=isNull(sTal.SmTalonID,0)
) s

left join oms_onco_N014 onco_14 ON
onco_14.onco_N014ID = sTal.rf_onco_N014ID and s.USL_TIP=1 /*
поднимаем по ссылке классификатор типов хирургического лечения для поля HIR_TIP */
left join oms_onco_N015 onco_15 ON
onco_15.onco_N015ID = sTal.rf_onco_N015ID and s.USL_TIP=2 /*
поднимаем по ссылке классификатор линий лекарственной терапии для поля LEK_TIP_L */
left join oms_onco_N016 onco_16 ON
onco_16.onco_N016ID = sTal.rf_onco_N016ID and s.USL_TIP=2 /*
поднимаем по ссылке классификатор циклов лекарственной терапии для поля LEK_TIP_V */
left join oms_onco_N017 onco_17 ON
onco_17.onco_N017ID = sTal.rf_onco_N017ID and s.USL_TIP in (3,4) /* поднимаем по ссылке
классификатор типов лучевой терапии для поля LUCH_TIP */
where u.DS_ONK in (1,2)

```

Выпускная квалификационная работа выполнена мной совершенно самостоятельно. Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

« \_\_\_ » \_\_\_\_\_ Г.

---

*(подпись)*

---

*(Ф.И.О.)*