|  |  |  |
| --- | --- | --- |
| СОГЛАСОВАНО | УТВЕРЖДАЮ | |
| Заместитель директора государственного бюджетного учреждения здравоохранения «Самарский областной медицинский информационно-аналитический центр» (МИАЦ) | | | Руководитель проектов общества с ограниченной ответственностью «Новософтим» (ООО «Новософтим») |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Лемаев  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.  М.П. | | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ю.С. Сильченко  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.  М.П. |
|  | | |  |

**ЕМИАС Самарской области**

**РУКОВОДСТВО АДМИНИСТРАТОРА**

Этап 2. Выполнение работ по развитию Системы в соответствии с функциональными требованиями п.4 Приложения №5 Функциональный компонент «Межведомственное взаимодействие с ГИС СОГ» Технического задания

Контракт от «17» декабря 2021 г. № 52/21-ДБУ

Содержание

[1 Термины и сокращения 3](#_Toc103959923)

[2 Введение 4](#_Toc103959924)

[3 Настройка шаблона внесения данных вызова на дом 5](#_Toc103959925)

[4 Добавление пункта меню 6](#_Toc103959926)

[5 Настройка интеграции / интеграционного пользователя 7](#_Toc103959927)

[6 Добавление анкеты 9](#_Toc103959928)

[7 Настройка пользовательских процедур 10](#_Toc103959929)

[7.1 Настройка пользовательской процедуры «SogAddByQuestionary» 10](#_Toc103959930)

[7.2 Настройка пользовательской процедуры SogSendHomeCall 11](#_Toc103959931)

[7.3 Настройка пользовательской процедуры SogAddByPalliative 12](#_Toc103959932)

[7.4 Настройка пользовательской процедуры SogAddByInability 14](#_Toc103959933)

[7.5 Настройка пользовательской процедуры SogSendHH 15](#_Toc103959934)

[7.6 Настройка пользовательской процедуры SogSendRecommendations 16](#_Toc103959935)

# Термины и сокращения

Термины и сокращения

|  |  |
| --- | --- |
| Термин, сокращение | Определение |
| ЕГИСЗ | Единая государственная информационная система здравоохранения Российской Федерации |
| ЕМИАС,  Система | Единая медицинская информационно-аналитическая система, обеспечивающая ведение централизованных информационных ресурсов в сфере здравоохранения Самарской области, информационное взаимодействие с ЕГИСЗ, организацию электронного медицинского документооборота |
| СДУ | Комплексная поддержка граждан пожилого возраста и инвалидов, направленная на обеспечение каждого человека, не полностью справляющегося с самостоятельным уходом, поддержкой самого высокого качества жизни с наивысшим возможным уровнем независимости, автономии, участия в деятельности, самореализации и человеческого достоинства. Главная ее цель – обеспечить людям, нуждающимся в уходе, социальное обслуживание и медицинскую помощь на дому, в полустационарной и стационарной формах, с привлечением патронажной службы и сиделок, а также поддержку родственного ухода. |

# Введение

Реестр «Система долговременного ухода» предназначена для организации порядка межведомственного взаимодействия медицинских организаций и организаций социального обслуживания, включающая синхронизацию информационных систем в соответствующих сферах, способы и объем передачи необходимой информации о состоянии гражданина его родственникам и организациям социального обслуживания.

В соответствии со статьей 28 Федерального закона № 442-ФЗ организация межведомственного взаимодействия включает в себя:

* обмен информацией, необходимой для предоставления социальных услуг и социального сопровождения;
* реализацию совместных действий учреждений здравоохранения и социального обслуживания в рамках индивидуальной программы предоставления социальных услуг (далее – ИППСУ);
* планирование совместной деятельности данных учреждений.

Целевые группы пациентов СДУ:

* гериатрические пациенты: граждане 65 лет и старше, в том числе живущие одинокие и одинокопроживающие;
* паллиативные пациенты;
* граждане с инвалидностью;
* граждане с психическими расстройствами, включая больных деменцией;
* маломобильные пациенты (необратимо и обратимо как результат успешной реабилитации);
* родственники, осуществляющие уход;
* граждане, оказывающие надомную помощь нуждающимся.

# Настройка шаблона внесения данных вызова на дом

Для настройки шаблона внесения данных вызова на дом выполните следующие действия:

Перейдите в пункт главного меню «Система» → «Настройка системных опций» → «Системные опции (Администратор)». Откроется форма настройки системных опций (Рисунок 1);

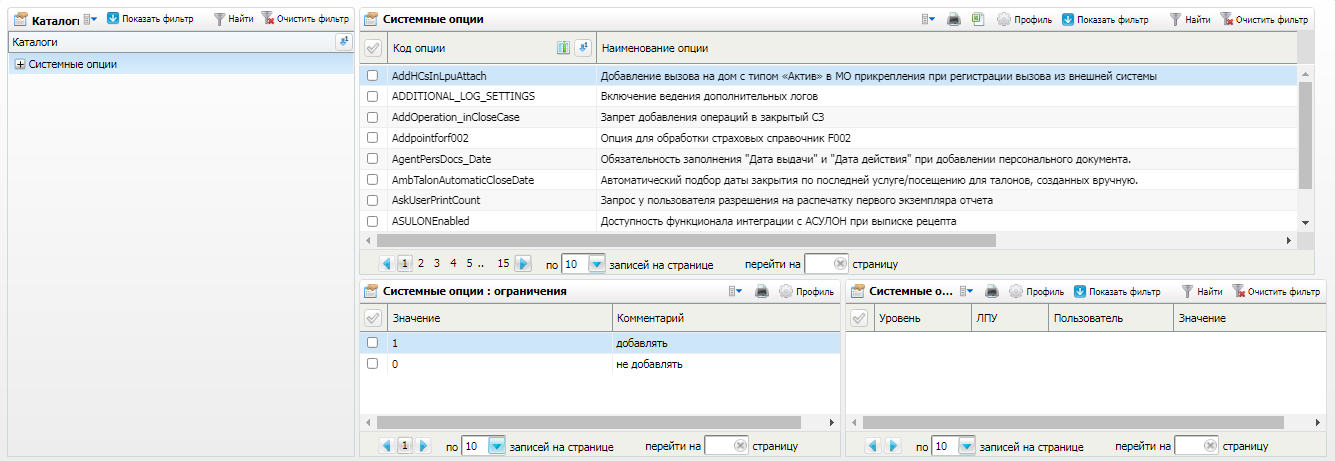


Рисунок 1 – Форма настройки системных опций

найдите СО «ServiceForCallToHome» – услуга вызова на дом по умолчанию.

Для интеграционного пользователя/МО указывается услуга с «расширенным шаблоном».

На «расширенный шаблон» добавляем вкладку с кодом «SOG\_SDU\_vkladka».

Добавьте поля, указанные в таблице 1.

Таблица 1 – поля для добавления СО

|  |  |
| --- | --- |
| Код поля | Наименование поля |
| ADD\_EXAMINATION | Назначение дополнительного обследования в связи с ухудшением состояния здоровья |
| ADD\_HOSPITAL | Госпитализация в медицинскую организацию |
| ADD\_HOSPITAL\_MO | Медицинская организация |
| EXTENSION\_COURSE | Продление курса лечебных мероприятий по назначению врача |
| EXTENSION\_COURSE\_DATE | Срок продления курса |
| REFUSAL | Оформление отказа от медицинской помощи |

# Добавление пункта меню

Для добавления пункта меню выполните следующие действия:

Перейдите в пункт главного меню «Система» → «Настройка главного меню». Откроется форма настройки главного меню (Рисунок 2).

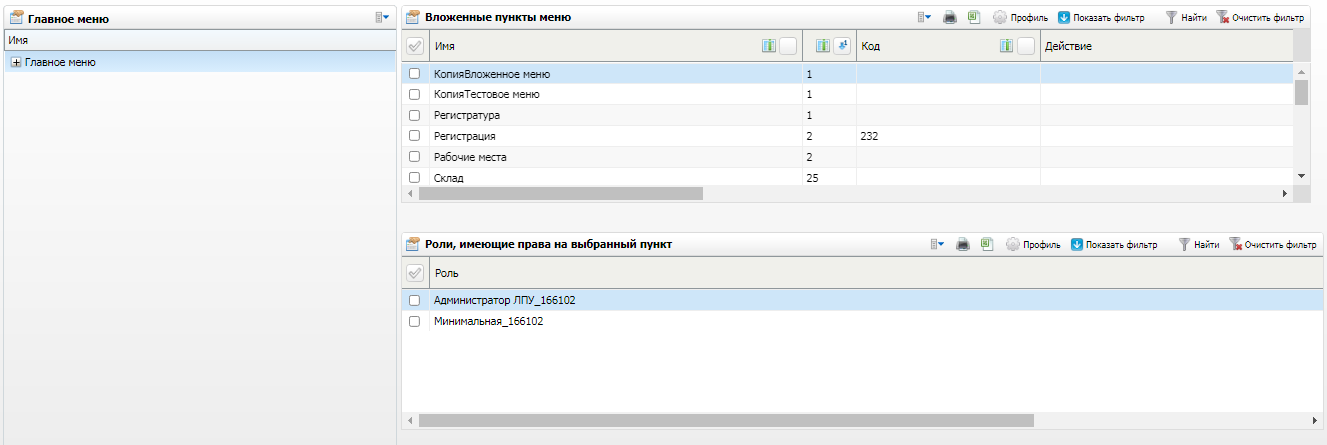


Рисунок 2 – Форма настройки главного меню

Выберите пункт контекстного меню «Добавить». Откроется окно добавления (Рисунок 3).

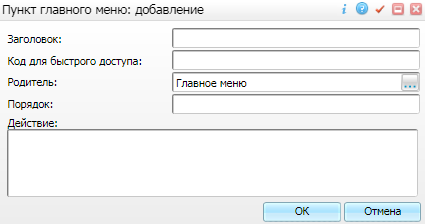


Рисунок 3 – Окно добавления пункта меню

Добавьте пункт меню по пути: «Учет» → «Нозологические регистры» → «Регистр "Социальное обслуживание граждан"» со следующими значениями:

Наименование пункта: «Регистр "Социальное обслуживание граждан"».

Действие: openD3Form('NR/social\_service/registry/main\_grid').

Нажмите на кнопку «Ок».

Добавьте в нижнем гриде роли, которым будет доступен пункт меню:

# Настройка интеграции / интеграционного пользователя

Для взаимодействия с СОГ СДУ необходимо настроить интеграционного пользователя. Для этого выполните следующие действия:

выберите пункт главного меню «Администратор» → «Пользователи\роли пользователей». Откроется форма настройки пользователей (Рисунок 4);

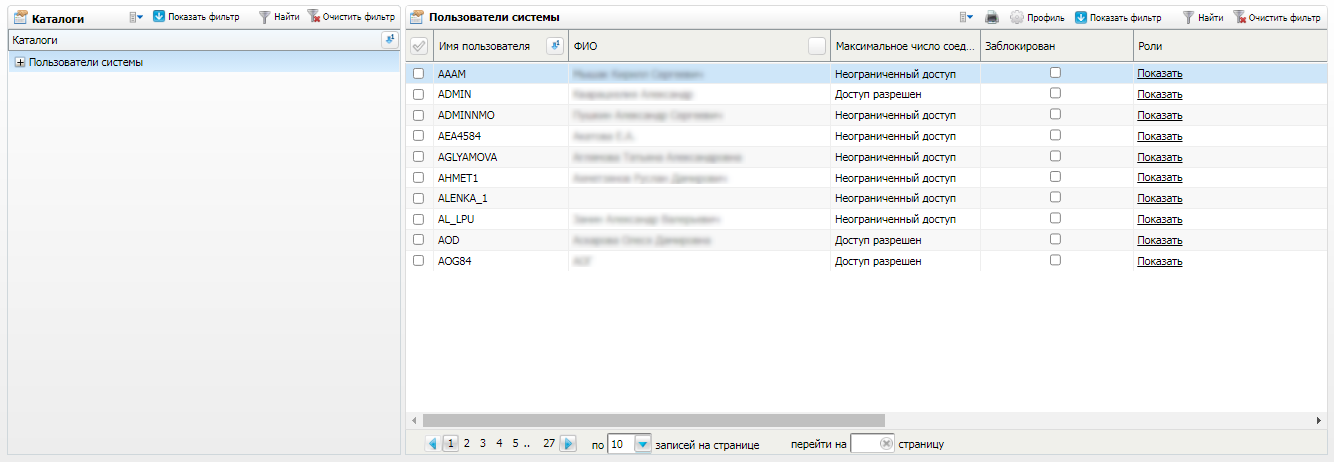


Рисунок 4 – Форма настройки пользователей

выберите пункт контекстного меню «Добавить» в блоке «Пользователи системы». Откроется окно добавления нового пользователя (Рисунок 5);

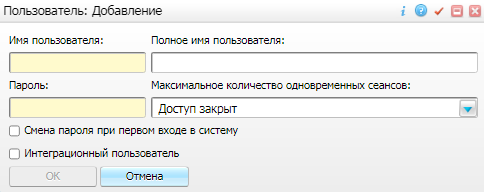


Рисунок 5 – Окно добавления пользователя

заполните поля в открывшемся окне;

нажмите на кнопку «Ок».

Интеграционному пользователю необходимо назначить права на все разделы. Для этого выполните следующие действия:

выделите строку с интеграционным пользователем в блоке «Пользователи системы»;

перейдите по ссылке «Показать» в столбце «Роли». Откроется окно «Роли и пользователи»;

выберите пункт контекстного меню «Добавить». Откроется окно добавления роли;

добавьте роль «Администратор»;

нажмите на кнопку «ОК» для сохранения внесенных данных

# Добавление анкеты

Добавьте анкету с кодом «SOG\_SDU\_asthenia» перейдя в пункт главного меню «Настройки» → «Настройка анкет».

Согласно анкете «Скрининг по шкале «Возраст не помеха» добавьте вопросы и ответы на вопросы «Да/Нет», где значение Да = 1, Нет = 0.

Добавить на шаблон врача вкладку «sog\_sdu\_anketa».

На вкладке укажите форму «Questionaries/questionaries\_tab» со скриптом «setVar('QUESTIONARY\_CODE', 'SOG\_SDU\_asthenia');», где указан код анкеты.

Скрипт для вкладки, чтобы отображать вкладку только у пациентов старше 65 лет:

setVar('QUESTIONARY\_CODE', 'SOG\_SDU\_asthenia');

getDataSet('DS\_QUEST\_TABS').addEventListener('afterrefresh', function () {

var name = null;

var data = document.querySelectorAll('div[cmptype="TabButton"]');

var i = 0;

while(i === (data.length - 1)) {

if (data[i].querySelector('div[class="TabCenter"]').innerText.search(/(65 и старше)/iu) != -1) {

name = data[i].getAttribute('name');

break;

}

i++;

}

if (name) {

setTabSheetVisibleByName(name, +getVar('AGN\_AGE') >= 65 ? 1 : 0);

}

});

# Настройка пользовательских процедур

Для настройки пользовательских процедур перейдите в пункт главного меню «Система» → «Пользовательские процедуры». Далее добавьте или настройте пользовательские процедуры.

## Настройка пользовательской процедуры «SogAddByQuestionary»

Настройте пользовательскую процедуру «SogAddByQuestionary» в соответствии с таблицей 2.

Таблица 2 – Параметры настройки пользовательской процедуры «SogAddByQuestionary»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код | Наименование | | Неименованный блок | |
| SogAddByQuestionary | Добавление в реестр СОГ СДУ по анкете | | declare nSUMM NUMBER(10,2); nAGENT NUMBER(17); rSOG\_SDU\_AGENT D\_SOG\_SDU\_AGENTS%rowtype; begin begin select t.SUMM, t.PID into nSUMM, nAGENT from D\_AGENT\_QUESTIONARIES t join D\_QUESTIONARIES t1 on t.QUESTIONARY = t1.ID where t.ID = <DOC\_ID> and t1.Q\_CODE = 'SOG\_SDU\_asthenia'; exception when NO\_DATA\_FOUND then return; end;  if nSUMM < 3 then return; end if;  begin select t.\* into rSOG\_SDU\_AGENT from D\_SOG\_SDU\_AGENTS t where t.AGENT = nAGENT and t.SOG\_DATE\_END is null; exception when NO\_DATA\_FOUND then null; end;  D\_PKG\_SOG\_SDU\_AGENTS.UPG(pnID => rSOG\_SDU\_AGENT.ID, pnLPU => <LPU>, pnAGENT => coalesce(rSOG\_SDU\_AGENT.AGENT, nAGENT), psSOG\_MAN\_ID => rSOG\_SDU\_AGENT.SOG\_MAN\_ID, pnSTATUS\_SOG => 1, pnASTHENIC\_PATIENT => nSUMM, pnSOCIAL\_STATUS => 0, pdSOG\_DATE\_BEGIN => null, pdSOG\_DATE\_END => null, pnREMOVE\_CAUSE => null, pnPAT\_GROUP => 1, psSOG\_REQUEST\_ID => rSOG\_SDU\_AGENT.SOG\_REQUEST\_ID); exception when OTHERS then null; end; | |
| **Раздел** | | **Действие** | **Перед** | **После** |
| AGENT\_QUESTIONARIES | | AGENT\_QUESTIONARIES\_UPDATE | 0 | 1 |

## Настройка пользовательской процедуры SogSendHomeCall

Настройте пользовательскую процедуру «SogSendHomeCall» в соответствии с таблицей 3.

Таблица 3 – Параметры настройки пользовательской процедуры «SogSendHomeCall»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код | | Наименование | | Неименованный блок |
| SogSendHomeCall | | Отправка в СОГ изменений по вызову на дом | | declare nID NUMBER(17); sERROR VARCHAR2(4000); sERRORS VARCHAR2(4000); nLOG\_ID NUMBER(17); clOUT\_VARS D\_CL\_SS; begin begin select max(hj.ID) into nID from D\_VISITS v join D\_HOME\_CALL\_JOUR hj on hj.DIRECTION\_SERVICE = v.PID join D\_EX\_SYSTEMS s on s.ID = hj.EX\_SYSTEM and s.S\_CODE = 'SogSdu' where v.ID = <DOC\_ID>; if nID is null then return; end if; D\_PKG\_IMP\_LOG.REINIT('SOG\_EXP\_HC\_' || to\_char(sysdate, '[dd.mm](http://dd.mm/).yyyy'), null, nLOG\_ID); INT\_P\_CALL\_MODULE(psFORM => 'Int/SogSdu/api', psMODULE => 'resultDepart', psIN\_VARS => 'ID='||nID, psERROR\_MSG => sERROR, psURL => 'local\_call', pnTIMEOUT => 30, pnRAISE\_TIMEOUT => 1, pclOUT\_VARS => clOUT\_VARS); select max(t.STR2) into sERROR from table(clOUT\_VARS) t where t.STR1 = '\_MESSAGE' and t.STR2 != 'Данные отправлены в СОГ СДУ.'; exception when OTHERS then sERROR := sqlerrm; end; if nLOG\_ID is not null then D\_PKG\_IMP\_LOG.ADD(nLOG\_ID, to\_char(nID) || case when sERROR is null then ' Ок' else ' Error' end, coalesce(sERROR, 'Ок')); end if; end; |
| **Раздел** | **Действие** | | **Перед** | **После** |
| VISITS | VISITS\_SAVE | | 0 | 1 |

## Настройка пользовательской процедуры SogAddByPalliative

Настройте пользовательскую процедуру «SogAddByPalliative» в соответствии с таблицей 4.

Таблица 4 – Параметры настройки пользовательской процедуры «SogAddByPalliative»

|  |  |  |  |
| --- | --- | --- | --- |
| Код | Наименование | Неименованный блок | |
| SogAddByPalliative | Добавление в реестр СОГ СДУ по паллиативному статусу | declare nAGENT NUMBER(17); rSOG\_SDU\_AGENT D\_SOG\_SDU\_AGENTS%rowtype; begin  begin select t.PID into nAGENT from D\_AGENT\_PALLIATIVE t where t.ID = <DOC\_ID> and coalesce(t.DATE\_E, sysdate) <= sysdate; exception when NO\_DATA\_FOUND then return; end;  begin select t.\* into rSOG\_SDU\_AGENT from D\_SOG\_SDU\_AGENTS t where t.AGENT = nAGENT and t.SOG\_DATE\_END is null; exception when OTHERS then null; end;  D\_PKG\_SOG\_SDU\_AGENTS.UPG(pnID => rSOG\_SDU\_AGENT.ID, pnLPU => <LPU>, pnAGENT => coalesce(rSOG\_SDU\_AGENT.AGENT, nAGENT), psSOG\_MAN\_ID => rSOG\_SDU\_AGENT.SOG\_MAN\_ID, pnSTATUS\_SOG => 1, pnASTHENIC\_PATIENT => null, pnSOCIAL\_STATUS => 0, pdSOG\_DATE\_BEGIN => null, pdSOG\_DATE\_END => null, pnREMOVE\_CAUSE => null, pnPAT\_GROUP => 2, -- Паллиативные пациенты psSOG\_REQUEST\_ID => rSOG\_SDU\_AGENT.SOG\_REQUEST\_ID); exception when OTHERS then null; end; | |
| **Раздел** | **Действие** | **Перед** | **После** |
| AGENT\_PALLIATIVE | AGENT\_PALLIATIVE\_INSERT | 0 | 1 |
| AGENT\_PALLIATIVE | AGENT\_PALLIATIVE\_UPDATE | 0 | 1 |

## Настройка пользовательской процедуры SogAddByInability

Настройте пользовательскую процедуру «SogAddByInability» в соответствии с таблицей 5.

Таблица 5 – Параметры настройки пользовательской процедуры «SogAddByInability»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Код | Наименование | | Неименованный блок | | |
| SogAddByInability | Добавление в реестр СОГ СДУ по инвалидности | | declare nAGENT NUMBER(17); rSOG\_SDU\_AGENT D\_SOG\_SDU\_AGENTS%rowtype; begin  begin select t.PID into nAGENT from D\_AGENT\_INABILITIES t join D\_AGENTS t1 on t1.ID = t.PID where t.ID = <DOC\_ID> and t.INABILITY\_GROUP in (1, 2) and t.INABILITY\_STATUS != 3 and coalesce(t.END\_DATE, sysdate) <= sysdate and D\_PKG\_DAT\_TOOLS.FULL\_YEARS(sysdate, t1.BIRTHDATE) >= 18; exception when NO\_DATA\_FOUND then return; end;  begin select t.\* into rSOG\_SDU\_AGENT from D\_SOG\_SDU\_AGENTS t where t.AGENT = nAGENT and t.SOG\_DATE\_END is null; exception when OTHERS then null; end;  D\_PKG\_SOG\_SDU\_AGENTS.UPG(pnID => rSOG\_SDU\_AGENT.ID, pnLPU => <LPU>, pnAGENT => coalesce(rSOG\_SDU\_AGENT.AGENT, nAGENT), psSOG\_MAN\_ID => rSOG\_SDU\_AGENT.SOG\_MAN\_ID, pnSTATUS\_SOG => 1, pnASTHENIC\_PATIENT => null, pnSOCIAL\_STATUS => 0, pdSOG\_DATE\_BEGIN => null, pdSOG\_DATE\_END => null, pnREMOVE\_CAUSE => null, pnPAT\_GROUP => 3, -- Граждане с инвалидностью psSOG\_REQUEST\_ID => rSOG\_SDU\_AGENT.SOG\_REQUEST\_ID); exception when OTHERS then null; end; | | |
| **Раздел** | | **Действие** | | **Перед** | **После** |
| AGENT\_INABILITIES | | AGENT\_INABILITIES\_INSERT | | 0 | 1 |
|  | | AGENT\_INABILITIES\_INSERT | | 0 | 1 |

## Настройка пользовательской процедуры SogSendHH

Настройте пользовательскую процедуру «SogSendHH» в соответствии с таблицей 6.

Таблица 6 – Параметры настройки пользовательской процедуры «SogSendHH»

|  |  |  |  |
| --- | --- | --- | --- |
| Код | Наименование | Неименованный блок | |
| SogSendHH | Отправка в СОГ информации о выписке | declare nID NUMBER(17); sERROR VARCHAR2(4000); sERRORS VARCHAR2(4000); nLOG\_ID NUMBER(17); clOUT\_VARS D\_CL\_SS; begin begin select max(hh.ID) into nID from D\_HOSP\_HISTORIES hh join D\_PERSMEDCARD p on p.ID = hh.PATIENT join D\_SOG\_SDU\_AGENTS ssa on ssa.AGENT = p.AGENT where hh.ID = <DOC\_ID> and hh.DATE\_OUT is not null and ssa.REMOVE\_CAUSE is null; if nID is null then return; end if; commit; D\_PKG\_IMP\_LOG.REINIT('SOG\_EXP\_HH\_' || to\_char(sysdate, '[dd.mm](http://dd.mm/).yyyy'), null, nLOG\_ID); INT\_P\_CALL\_MODULE(psFORM => 'Int/SogSdu/api', psMODULE => 'sendDischarge', psIN\_VARS => 'ID='||nID, psERROR\_MSG => sERROR, psURL => 'local\_call', pnTIMEOUT => 30, pnRAISE\_TIMEOUT => 1, pclOUT\_VARS => clOUT\_VARS); select max(t.STR2) into sERROR from table(clOUT\_VARS) t where t.STR1 = '\_MESSAGE' and t.STR2 != 'Данные отправлены в СОГ СДУ.'; exception when OTHERS then sERROR := sqlerrm; end; if nLOG\_ID is not null then D\_PKG\_IMP\_LOG.ADD(nLOG\_ID, to\_char(nID) || case when sERROR is null then ' Ок' else ' Error' end, coalesce(sERROR, 'Ок')); end if; end; | |
| **Раздел** | **Действие** | **Перед** | **После** |
| HOSP\_HISTORIES | HOSP\_HISTORIES\_CLOSE | 0 | 1 |

## Настройка пользовательской процедуры SogSendRecommendations

Настройте пользовательскую процедуру «SogSendRecommendations» в соответствии с таблицей 7.

Таблица 7 – Параметры настройки пользовательской процедуры «SogSendRecommendations»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Код | Наименование | | Неименованный блок | | |
| SogSendRecommendations | Отправка в СОГ изменений по рекомендациям | | declare nID NUMBER(17); sERROR VARCHAR2(4000); sERRORS VARCHAR2(4000); nLOG\_ID NUMBER(17); clOUT\_VARS D\_CL\_SS; begin begin select max(ssma.ID) into nID from D\_SOG\_SDU\_MED\_ADVICE ssma join D\_V\_SOG\_SDU\_AGENTS ssa on ssa.ID = ssma.PID where ssma.ID = <DOC\_ID> and ssa.SOG\_MAN\_ID is not null and ssma.IS\_LAST\_ADVICE = 1; if nID is null then return; end if; commit; D\_PKG\_IMP\_LOG.REINIT('SOG\_EXP\_ADV\_' || to\_char(sysdate, '[dd.mm](http://dd.mm/).yyyy'), null, nLOG\_ID); INT\_P\_CALL\_MODULE(psFORM => 'Int/SogSdu/api', psMODULE => 'updateAdvice', psIN\_VARS => 'ID='||nID, psERROR\_MSG => sERROR, psURL => 'local\_call', pnTIMEOUT => 30, pnRAISE\_TIMEOUT => 1, pclOUT\_VARS => clOUT\_VARS); select max(t.STR2) into sERROR from table(clOUT\_VARS) t where t.STR1 = '\_MESSAGE' and t.STR2 != 'Данные отправлены в СОГ СДУ.'; exception when OTHERS then sERROR := sqlerrm; end; if nLOG\_ID is not null then D\_PKG\_IMP\_LOG.ADD(nLOG\_ID, to\_char(nID) || case when sERROR is null then ' Ок' else ' Error' end, coalesce(sERROR, 'Ок')); end if; end; | | |
| **Раздел** | | **Действие** | | **Перед** | **После** |
| SOG\_SDU\_MED\_ADVICE | | SOG\_SDU\_MED\_ADVICE\_UPDATE | | 0 | 1 |
|  | | SOG\_SDU\_MED\_ADVICE\_DELETE | | 1 | 0 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в доку- менте | Номер доку- мента | Входящий номер сопроводи- тельного документа и дата | Под- пись | Дата |
| изменен- ных | заменен- ных | новых | аннулиро- ванных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |