



# Интеграция с SMS и мессенджерами

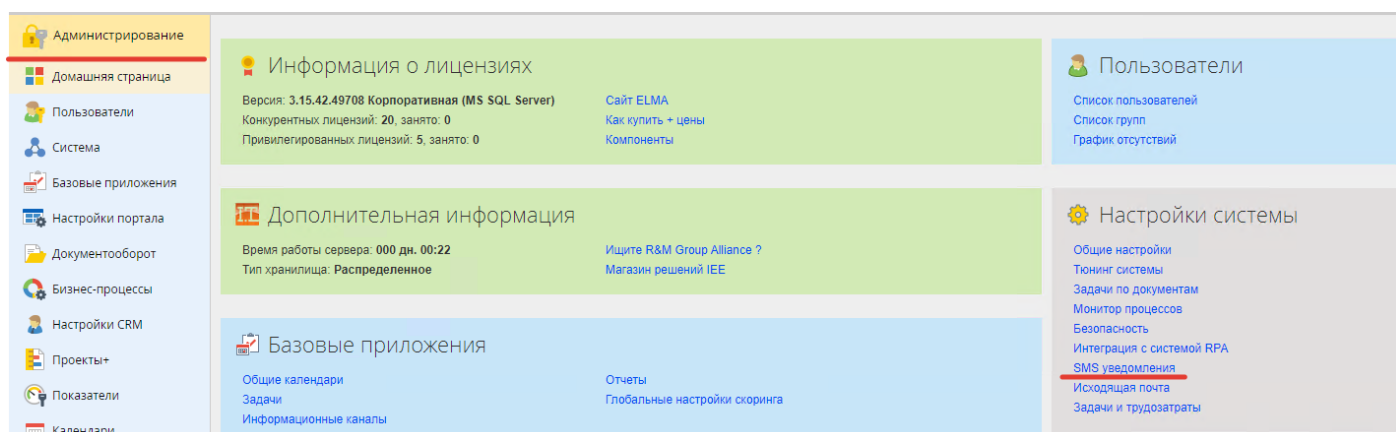
Модуль интеграции ядра системы с операторами SMS и мессенджерами

- Настройки
  - Настройки чат бота Telegram
- Примеры сценариев
  - Отправка сообщения в чат бот Telegram
  - Задача согласования документа в Telegram
  - Обычная задача бизнес процесса в Telegram
  - Возможность ввода комментария к документу
  - Комментарии к согласованию и ознакомлению
  - Задача ознакомления с документом в Telegram
  - Отключение вывода описания задачи в Telegram

# Настройки

# Настройки чат бота Telegram

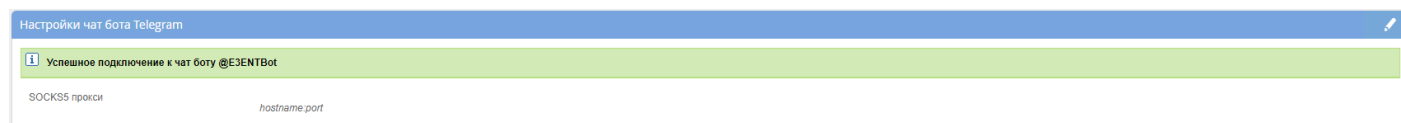
Для настройки подключения к чат боту Telegram нужно пройти по ссылке  
**Администрирование - (секция Настройки системы) SMS уведомления:**



Необходимо указать API Token чат бота, полученный от [@BotFather](#):

The image shows a dialog box titled 'Настройки чат бота Telegram'. It has two input fields. The first field is labeled 'Токен' (Token) and contains the text 'p053'. Below it is a hint: 'API токен, полученный от @BotFather'. The second field is labeled 'SOCKS5 прокси' (SOCKS5 proxy) and is empty. Below it is a hint: 'hostname:port'. At the bottom right of the dialog are two buttons: 'Сохранить' (Save) and 'Отмена' (Cancel).

В случае успешного соединения, будет отображена следующая информация:



Если токен указан не верно, будет выдана соответствующая ошибка:

Настройки чат бота Telegram

Неверно указан токен чат бота

SOCKS5 проксиhostname.port

Если токен не указан, то интеграция отключена:

Настройки чат бота Telegram

Не указан токен чат бота

SOCKS5 проксиhostname.port

Доступно только в Корпоративной редакции

Сервис чат бота Telegram реализован в виде кворум сервиса. Если просмотр настроек производится на пассивном узле кластера, то будет выдано следующее предупреждение:

Настройки чат бота Telegram

Пассивный узел кластера

SOCKS5 проксиhostname.port

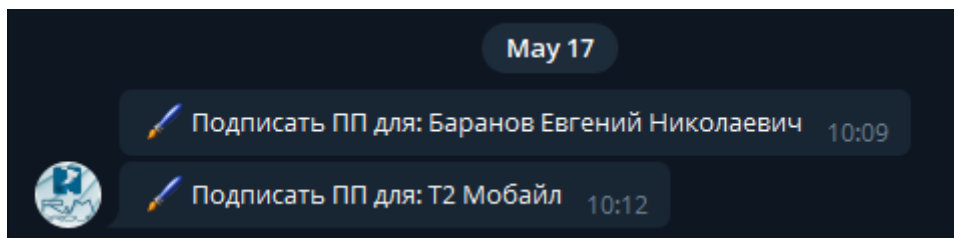
# Примеры сценариев

# Отправка сообщения в чат бот Telegram

При создании задачи в бизнес процессе можно отправить пользователю сообщение в чат бот Telegram:

```
public override void OnTaskCreate (ITaskBase task, P_SupplierAdvPayment context)
{
    IWorkflowTaskBase wtask = task as IWorkflowTaskBase;
    // {3d701012-72a0-4f87-8b27-5405c91d7181} - Uid задачи на схеме
    if (wtask != null && wtask.WorkflowBookmark.ElementUid.Equals (new Guid ("{3d701012-72a0-4f87-8b27-5405c91d7181}")) && wtask.Executor.MobilePhone != null) {
        try {
            GateManager.CurrentGate.SendMessage (wtask.Executor.MobilePhone, SR.Т["Подписать ПП для: {0}", context.Supplier.Name]);
        }
        catch {
            // ignored
        }
    }
}
```

В Telegram придет следующее сообщение:



Так же возможна отправка напрямую через метод **SMSHelper.ChatBotTelegram.SendTextMessage(user, message)**

# Задача согласования документа в Telegram

При постановке в бизнес процессе задачи согласования документа, можно ее продублировать в Telegram:

```
public override void OnTaskCreate (ITaskBase task, P_SupplierAdvPayment context)
{
    // d8ef716f-f819-4578-9cde-698ba71524b1 - Uid задачи на диаграмме
    SMSHelper.ChatBotTelegram.LinkWorkflowTask(new Guid("d8ef716f-f819-4578-9cde-698ba71524b1"), task,
    model =>
    {
        // 542bdff7-8d93-4946-bbc3-e2941d210daf - Uid перехода Согласовано
        model.Buttons.Add(new TelegramButton { Name = "Согласовано", Emoji = "👍", ConnectorUid = new
        Guid("542bdff7-8d93-4946-bbc3-e2941d210daf") });

        // c17b1064-0dbf-4372-b74d-38c84af7d05a - Uid перехода Не согласовано
        model.Buttons.Add(new TelegramButton { Name = "Отказано", Emoji = "👎", ConnectorUid = new
        Guid("c17b1064-0dbf-4372-b74d-38c84af7d05a") });

        // Счет
        if (context.Invoice != null)
            model.Document = new TelegramDocument { Name = context.Invoice.Name, File =
            context.Invoice.CurrentVersion.File };

        // Дополнительная информация
        model.Details.Add("Покупатель: " + context.Invoice.IEECustomer.Name);

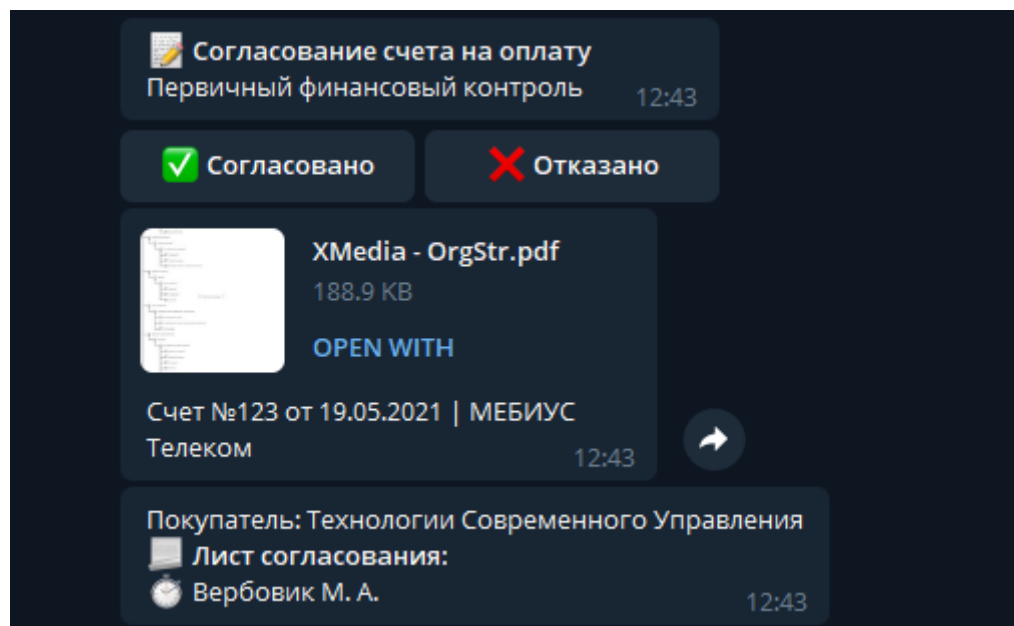
        // Лист согласования документа
        SMSHelper.ChatBotTelegram.AddApprovalList(model, context.Invoice);
    });
}
```

Необходимо добавить:

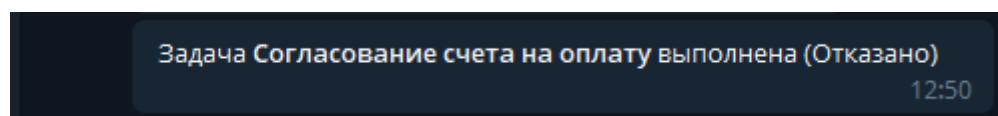
```
using ITino.ELMA.SMS.Helpers;
```

```
using ITino.ELMA.SMS.Models;
```

В результате в Telegram придет блок сообщений с кнопками:



При выборе действия "Отказано" задачу будет завершена и в бизнес процессе произойдет переход "Не согласовано":





# Обычная задача бизнес процесса в Telegram

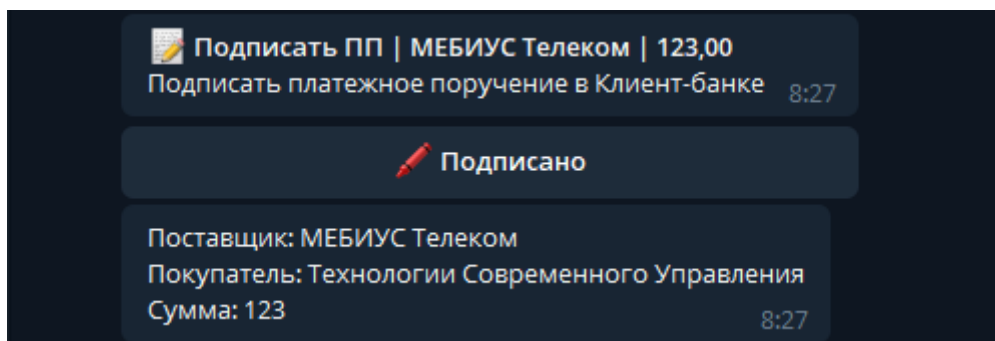
При выполнении обычной задачи бизнес процесса ее можно продублировать в Telegram, например задача подписания документа (не docflow):

```
public override void OnTaskCreate (ITaskBase task, P_SupplierAdvPayment context)
{
    // 1c2cbe8f-73b7-4618-930a-aeefe4ae06ff - Uid задачи на схеме
    SMSHelper.ChatBotTelegram.LinkWorkflowTask(new Guid("1c2cbe8f-73b7-4618-930a-aeefe4ae06ff"), task,
    model =>
    {
        // 5d682c20-4a47-433b-95c6-a08f97d56e68 - Uid перехода на схеме
        model.Buttons.Add(new TelegramButton { Name = "Подписано", Emoji = "🖋️", ConnectorUid = new
        Guid("5d682c20-4a47-433b-95c6-a08f97d56e68") });
    }
    // Файл (версия) документа
    model.Document = new TelegramDocument { File = context.PaymentOrder.File };
});
}
```

Необходимо добавить:

```
using ITino.ELMA.SMS.Helpers;
using ITino.ELMA.SMS.Models;
```

В Telegramt будет сформирован блок сообщений:



После нажатия "Подписано" задача будет завершена и бизнес процесс пойдет дальше:

Задача Подписать ПП | МЕБИУС Телеком | 123,00 выполнена (Подписано)

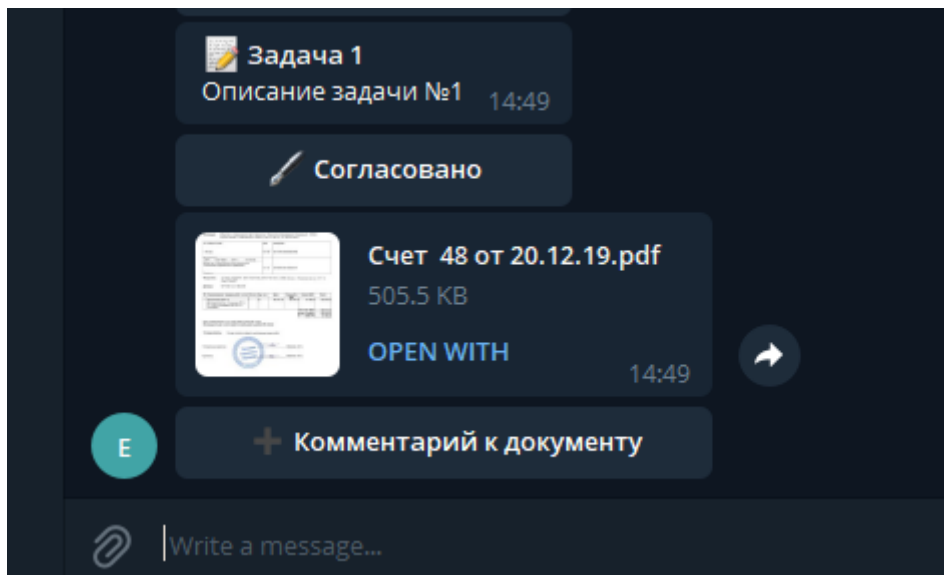
8:27

# Возможность ввода комментария к документу

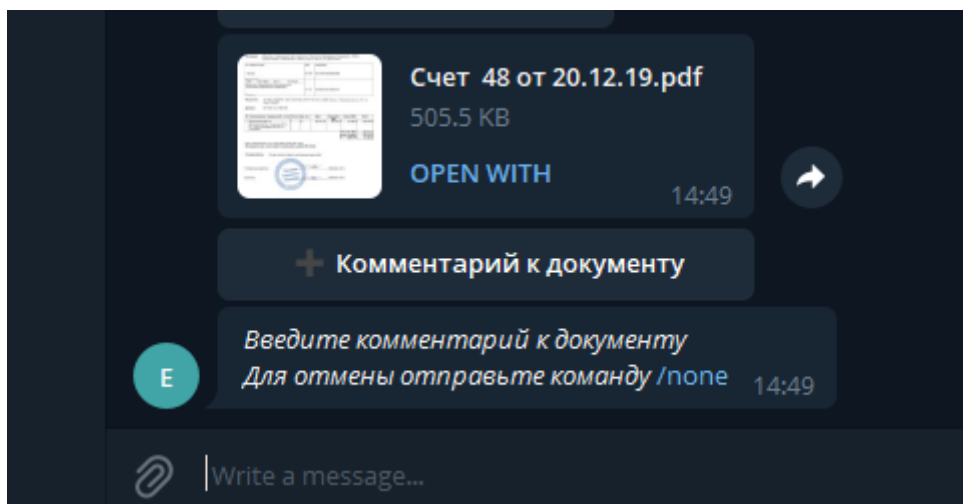
Для реализации возможности ввода комментария к документу в процессной задаче, необходимо указать его Id:

```
// Просто задача
SMSHelper.ChatBotTelegram.LinkWorkflowTask(new Guid("d07a5dc2-e7d8-457e-861a-ed2daeb7e199"), task,
model =>
{
    model.Buttons.Add(new TelegramButton {Name = "Согласовано", Emoji = "👍", ConnectorUid = new
Guid("9de8cf05-e5a1-4cdb-9e5a-7a2af6d22fa3")});
    // Указываем Id документа для возможности ввода комментария
    model.Document = new TelegramDocument { File = context.Invoice.CurrentVersion.File, Id = context.Invoice.Id
};
});
```

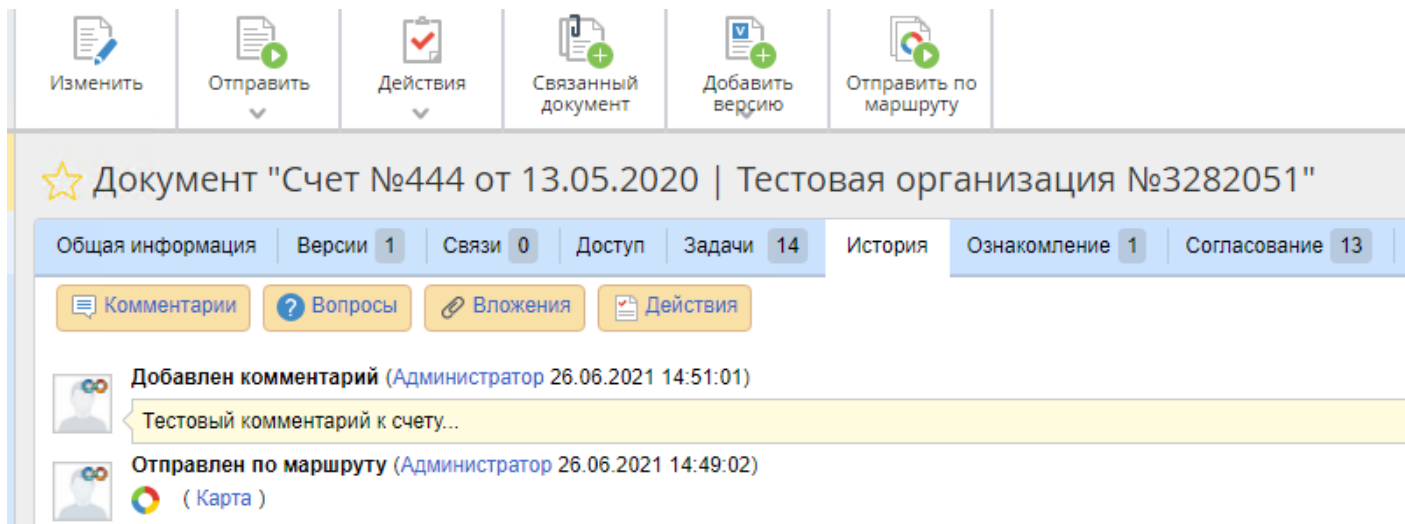
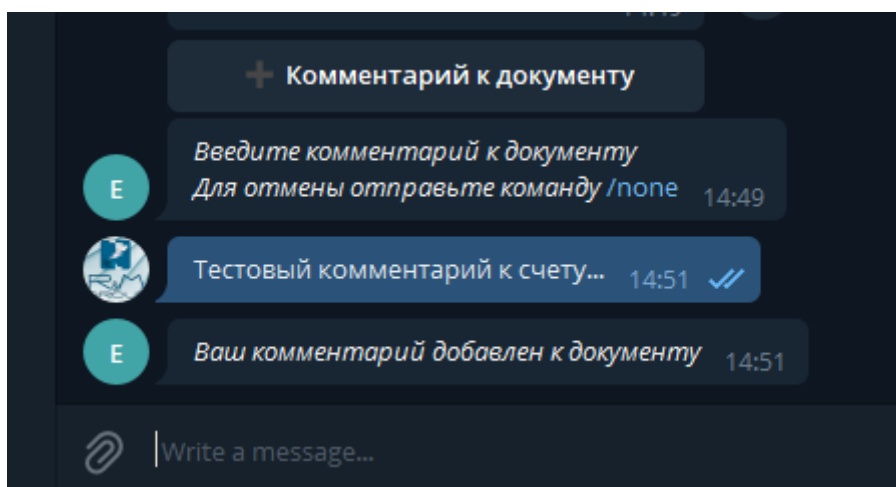
В результате создания задачи будет добавлена кнопка "+ Комментарий к документу":



При нажатии на указанную кнопку бот выдаст подсказку:



После ввода комментария и отправки его боту, получим подтверждение и комментарий в самом документе:



Если нужно отменить ввод комментария, необходимо отправить команду **/none**:



Введите комментарий к документу  
Для отмены отправьте команду /none 14:51



/none 14:52 ✓✓



Ввод комментария к документу отменен 14:52



Write a message...

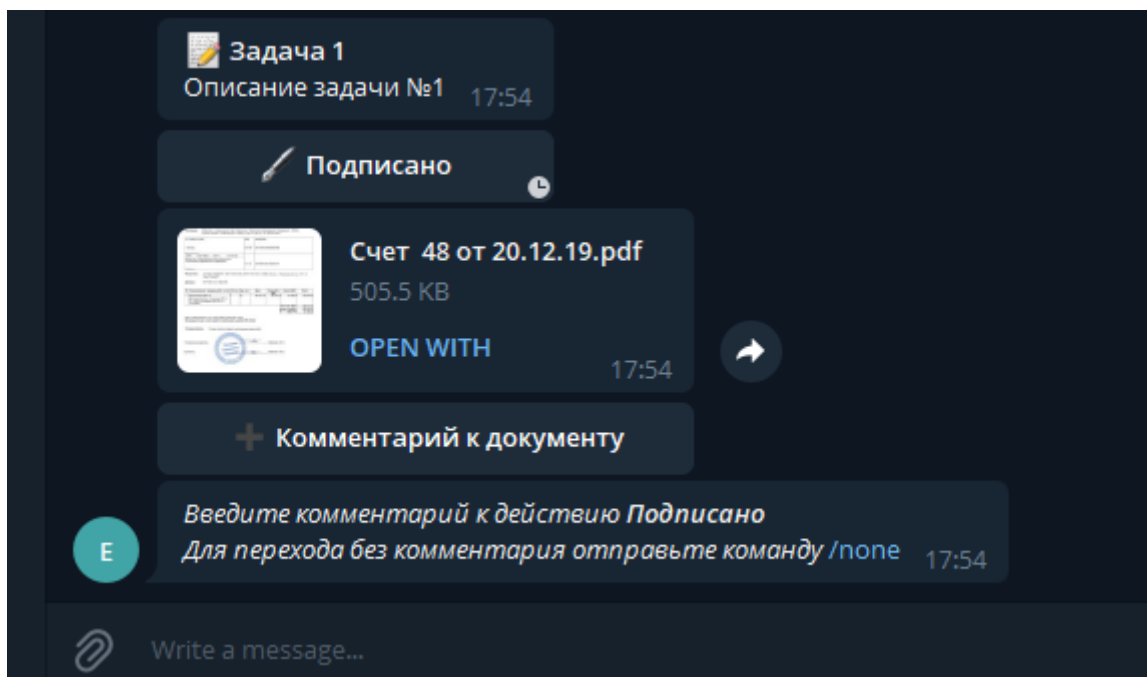
# Комментарии к согласованию и ознакомлению

При выполнении задач по согласованию или ознакомлению документа, возможно указать вариант ввода комментария. Есть 3 варианта:

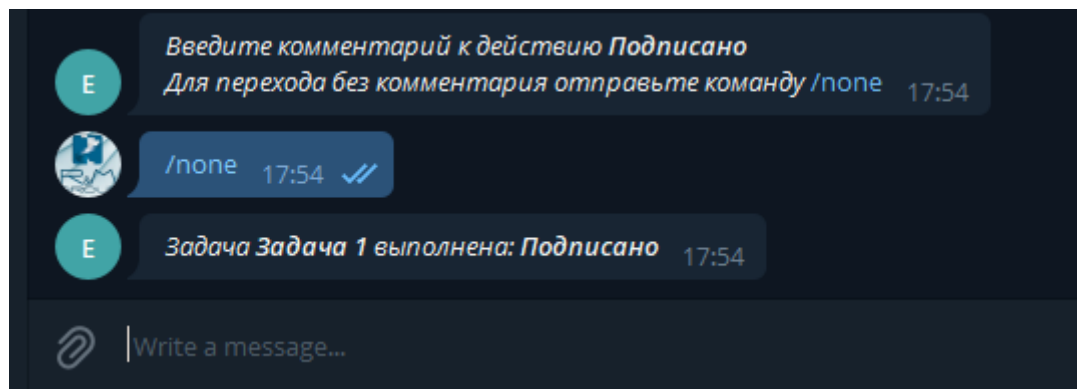
- Не вводить
- Опционально
- Обязательно

```
// По умолчанию AddCommentOption = EOAddCommentOption.None
model.Buttons.Add(new TelegramButton { Name = "Согласовано", Emoji = "👍", ConnectorUid = new
Guid("9d0c9a7d-5db0-4c3e-b7b3-e8083fa0dd27"), AddCommentOption = EOAddCommentOption.Optional });
model.Buttons.Add(new TelegramButton { Name = "Отказано", Emoji = "👎", ConnectorUid = new
Guid("b8f9eca4-6a1a-408e-af99-3a546499e191"), AddCommentOption = EOAddCommentOption.Required });
```

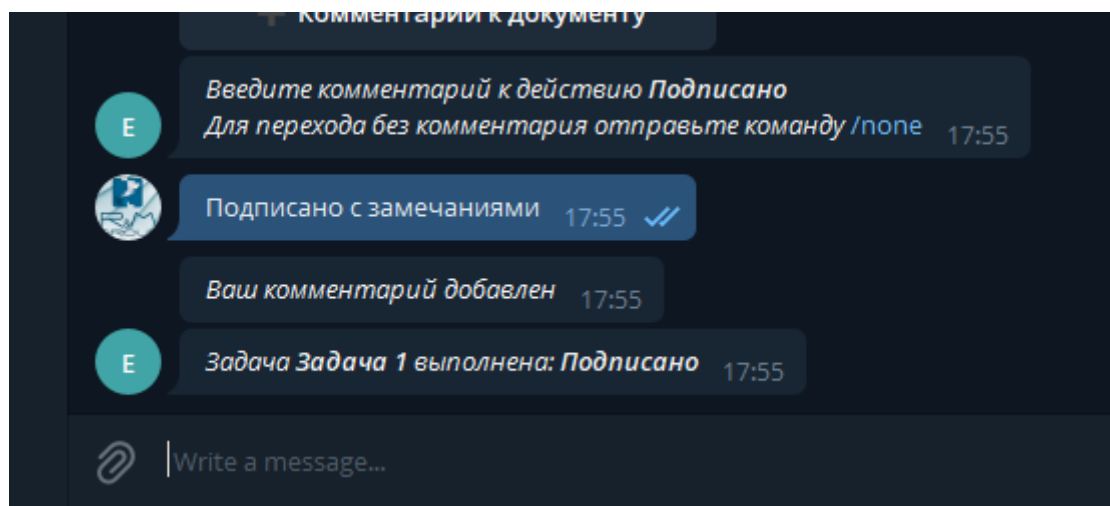
Для опционального комментария чат бот пришлет следующее сообщение:



И если будет введена команда **/none**, действие будет выполнено без фиксации комментария:



Для обязательного комментария действие не будет выполнено, пока не будет введен текст, а по факту его ввода будет выдано соответствующее уведомление:



# Задача ознакомления с документом в Telegram

Функционал аналогичен [Задаче согласования документа в Telegram](#).

При этом можно сформировать лист ознакомления следующим образом:

```
SMSHelper.ChatBotTelegram.LinkWorkflowTask(new Guid("932cffc4-7f2e-4570-81f4-42fc25a7ea28"), task,
model =>
{
    model.Buttons.Add(new TelegramButton { Name = "Ознакомлен", Emoji = "👍", ConnectorUid = new
Guid("8a771e9c-b538-4b3e-8815-d57107fd8dfd") });

    if (context.Invoice != null)
        model.Document = new TelegramDocument { Name = context.Invoice.Name, File =
context.Invoice.CurrentVersion.File };
}

// Добавить лист ознакомления по документу
SMSHelper.ChatBotTelegram.AddAcquaintanceList(model, context.Invoice);
});
```



# Отключение вывода описания задачи в Telegram

По умолчанию, в чат выводится после названия задачи ее описание, которое заполнено в Дизайнере. Если описание слишком большое, либо его не надо выводить в чат, необходимо в модели при создании задачи указать следующее:

```
model.DisableTaskDescription = true;
```

Ссылка на пример сценария: [Обычная задача бизнес процесса в Telegram](#)