


???????

- [Обработка входящего Договора](#)
- [Обработка исходящего Договора с физ. лицом](#)
- [Отправка документа сценарием на подписание в Диадок](#)
- [Добавление скан-копии](#)
- [Автоматический переход в процессной задаче](#)

????????? ??????????
?????????

В веб дизайнере ELMA4 (в ELMA3 обычном дизайнере) реализуем Глобальный модуль, где определяем нужную точку расширения, например, **ReceivableContract**.

 **Дизайнер**

Сценарии ▾

Поиск

Все сценарии R&M HMS ✕

Глобальный модуль "R&M HMS"

Настройки

Название *

R&M HMS

Пространство имен

HMS.Configuration

Описание

Структура модуля

▾ Корневая папка модуля

▸ Contracts

▾ Diadoc

SendableRNMOutContractSubLic

ReceivableContract

DiadocIntegration

ReceivableAdditionalAgreement

SendableRNMOutContractInd

SendableRNMOutContractLic

Реализация:

```
using System;
using EleWise.ELMA.ComponentModel;
using EleWise.ELMA.Documents.Contracts.Models;
using EleWise.ELMA.Model.Services;
using ITino.ELMA.Documents.Diadoc.Components;
using ITino.ELMA.Documents.Diadoc.Models;

namespace HMS.Configuration.Diadoc
{
```

```

/// <inheritdoc />
[Component]
public class ReceivableContract : ReceivableDiadocDocument
{
    /// <inheritdoc />
    public override bool Enabled
    {
        /// Точка расширения включена и обрабатывается
        get { return true; }
    }

    /// <inheritdoc />
    public override DiadocDocumentType DocumentType
    {
        /// Какой тип документа Диадока обрабатываем
        get { return DiadocDocumentType.Contract; }
    }


    /// <inheritdoc />
    public override Guid TypeUid
    {
        /// К какому типу документа в ELMA привязываем
        get { return InterfaceActivator.UID<IContractIncoming>(); }
    }
}
}

```

Атрибут [Component] обязателен.

????????? ??????????
????????? ? ????. ??????

В веб дизайнере ELMA4 (в ELMA3 обычном дизайнере) реализуем Глобальный модуль, где определяем нужную точку расширения, например, **SendableRNMOutContractInd**.

Дизайнер

Сценарии ▾

Поиск

Все сценарииR&M HMS ✕

Глобальный модуль "R&M HMS"

Настройки

Название *

R&M HMS

Пространство имен

HMS.Configuration

Описание

Структура модуля

Корневая папка модуля

Contracts

Diadoc

SendableRNMOutContractSubLic

ReceivableContract

DiadocIntegration

ReceivableAdditionalAgreement

SendableRNMOutContractInd

SendableRNMOutContractLic

Products

Реализация:

```
using System;
using EleWise.ELMA.ComponentModel;
using EleWise.ELMA.ConfigurationModel;
using EleWise.ELMA.Documents.Models;
using EleWise.ELMA.CRM.Models;
using EleWise.ELMA.Model.Services;
using EleWise.ELMA.Services;
```

```

using ITino.ELMA.Documents.Diadoc.Components;
using ITino.ELMA.Documents.Diadoc.Exceptions;
using ITino.ELMA.Documents.Diadoc.Managers;
using ITino.ELMA.Documents.Diadoc.Models;
using ITino.ELMA.Documents.Diadoc.Services;

namespace HMS.Configuration
{
    /// <inheritdoc />
    [Component]
    public class SendableRNMOutContractInd : SendableDiadocDocument
    {
        /// <inheritdoc />
        public override bool Enabled
        {
            /// Точка расширения включена и обрабатывается
            get { return true; }
        }

        /// <inheritdoc />
        public override DiadocDocumentType DocumentType
        {
            /// Какой тип документа Диадока обрабатываем
            get { return DiadocDocumentType.Contract; }
        }

        /// <inheritdoc />
        public override Guid TypeUId
        {
            /// К какому типу документа в ELMA привязываем
            get { return InterfaceActivator.UID<RNMOutContractInd>(); }
        }

        public override IContractor GetSender(IDocument doc)
        {
            /// Получить отправителя документа
            var contract = doc as RNMOutContractInd;
            return contract != null ? contract.Contract : null;
        }
    }
}

```

```

    /// <inheritdoc />
    public override EODiadocCanSendStatus CanSendDocument(IDocument doc)
    {
        /// Проверить, можно ли отправить документ в Диадок
        if (doc == null)
            return EODiadocCanSendStatus.DocIsNull;

        var service = Locator.GetServiceNotNull<IDiadocService>();
        var contract = doc as RNMOOutContractInd;
        if (contract != null)
            return service.CanSendDocument(contract.Contractor, contract.Executor);
        return base.CanSendDocument(doc);
    }

    /// <inheritdoc />
    public override ICODiadocDocument Send(IDocument doc, Guid? departmentId, Guid?
userId, string comment = null)
    {
        /// Отправить документ в Диадок на подписание нами
        var contract = doc as RNMOOutContractInd;
        if (contract == null)
            throw new DiadocException("Договор не может быть null");

        var edata = DiadocDocumentManager.Instance.CheckContractors(contract.Contractor,
contract.Executor, null, null);
        var result = edata.Service.SendContract(edata.FormBoxId, edata.ToBoxId, doc,
contract.Number, contract.ContractDate.Value, "ГПХ/Физ.лицо/Самозанятый", null, departmentId,
userId, comment);
        return DiadocDocumentManager.Instance.SendDocHistory(result, doc,
contract.Contractor, contract.Executor, contract.Number, contract.ContractDate);
    }
}

```

Атрибут [Component] обязателен.

???????? ???? ?????


????????? ?? ?????????? ?

??????

TODO

???????????? ????-??????

В веб дизайнере ELMA4 (в ELMA3 обычном дизайнере) реализуем Глобальный модуль, где определяем нужную точку расширения, например, **DiadocIntegration**.

 **Дизайнер**

Сценарии ▾

Поиск

Все сценарии

R&M HMS ✕

Глобальный модуль "R&M HMS"

Настройки

Название *

R&M HMS

Пространство имен

HMS.Configuration

Описание

Структура модуля

Корневая папка модуля

Contracts

Diadoc

SendableRNMOutContractSubLic

ReceivableContract

DiadocIntegration

ReceivableAdditionalAgreement

Реализация:

```
using EleWise.ELMA.ComponentModel;
using EleWise.ELMA.Documents.Models;
using EleWise.ELMA.Files;
using ITino.ELMA.Documents.Diadoc.Extensions;
using ITino.ELMA.Documents.Models;
using ITino.ELMA.Documents.Managers;

namespace HMS.Configuration.Diadoc
{
    [Component]
```



```
public class DiadocIntegration : BaseDiadocIntegration
{
    public override void AddScanCopy(IDocument document, BinaryFile file, string comment =
null)
    {
        /// Реализация собственной логики добавления скан-копии к кокументу, например:
        DocumentExtManager.Instance.AddScanCopy(document as IDODocument, file,
checkUseScanCopy: false);
    }
}
}
```

Атрибут [Component] обязателен.

???????????????? ???? ?
???????????? ???? ?

Начиная с 4.0.28 и 3.15.48

При обработке документов Диадока в процессе может потребоваться отслеживать состояние подписания (как для входящих, так и исходящих). Чтобы задача автоматически выполнялась по нужному переходу при наступлении события по документу в Диадоке, необходимо реализовать в сценарии процесса следующий код:

```
using ITino.ELMA.Diadoc.Helpers;

public override void OnTaskCreate(ITaskBase task, Context context)
{
    // Где,
    // 13b27a4a-b94f-4bc9-9390-d7ff4fcb58bd - идентификатор задачи на диаграмме процесса
    // 309f52b9-087d-4971-9fc0-9cbffc61c737 - идентификатор перехода в случае, если документ
    Подписан
    task.LinkDiadocEvents(new Guid("13b27a4a-b94f-4bc9-9390-d7ff4fcb58bd"),
        context.DiadocDocument, new Guid("309f52b9-087d-4971-9fc0-9cbffc61c737"));
}
```

Указанный пример обрабатывает событие **Подписан**. При необходимости можно указать переходы для событий **Отклонен** и **Удален**.