

Лид

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

Содержание

Лид	4
Создать пользовательские напоминания и уведомления	4
Создать пользовательское напоминание о дате актуализации продажи в лиде	4
Описание примера	4
Исходный код	4
Алгоритм реализации примера	4

Лид



Сложный

Создать пользовательские напоминания и уведомления

В версии 7.12.0 переработан механизм отправки уведомлений и напоминаний.

Ранее для отправки пользовательского уведомления требовалось:

- Создать класс, реализующий интерфейс `INotificationProvider` или наследующий абстрактный класс `BaseNotificationProvider`.
- В созданный класс добавить логику для выбора пользовательских напоминаний приложением.
- Зарегистрировать класс в таблице `NotificationProvider`.

При этом уведомления отправлялись раз в минуту, вызывая все классы из таблицы `NotificationProvider`.

Начиная с версии 7.12.0 достаточно создать уведомление или напоминание с нужными параметрами. После этого приложение либо отправит уведомление мгновенно, либо отобразит напоминание в указанное время.

Чтобы отправлять пользовательские напоминания:

1. В пользовательском пакете [создайте схему \[Исходный код \]](#), в которой определите класс для формирования текста напоминания и всплывающего окна. Класс должен реализовывать интерфейс `IRemindingTextFormer` (объявлен в схеме `IRemindingTextFormer` пакета `Base`).
2. Заместите нужный объект (например, объект [Лид]) и задайте в нем логику отправки напоминания.
3. Заместите схему вкладки напоминаний `ReminderNotificationsSchema` для отображения напоминаний по требуемому объекту.

Создать пользовательское напоминание о дате актуализации продажи в лиде



Средний

Описание примера

Создать пользовательское напоминание о дате актуализации продажи в лиде. Дата должна быть указана в поле [Дата следующей актуализации] на вкладке [Информация о сделке].

Исходный код

Пакет с реализацией примера для Sales Creatio, enterprise edition можно скачать по [ссылке](#).

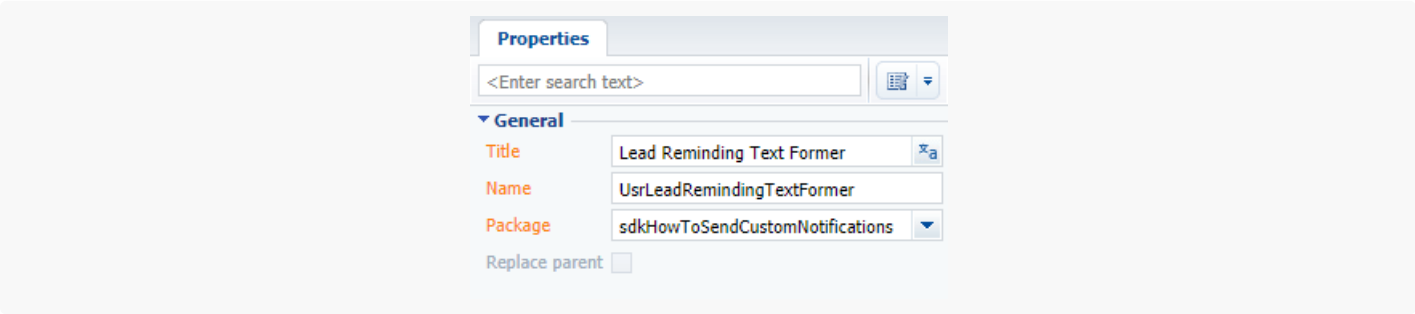
Алгоритм реализации примера

1. Создать класс для формирования текста напоминания и всплывающего окна

1. Создайте в пользовательском пакете схему [*Исходный код*] (см. "[Создать схему \[Исходный код \]](#)"). Установите для нее следующие значения свойств (рис. 1):

- [Заголовок] ([Title]) — "Конструктор текста напоминаний лида" ("Lead Reminding Text Former").
- [Название] ([Name]) — "UsrLeadRemindingTextFormer".

Рис. 1. — Свойства схемы [*Исходный код*]



2. Используя контекстное меню вкладки [*Структура*] ([*Structure*]), добавьте в схему две локализуемые строки (рис. 2). Свойства локализуемых строк приведены в таблице 1.

Рис. 2. — Добавление локализуемых строк в схему

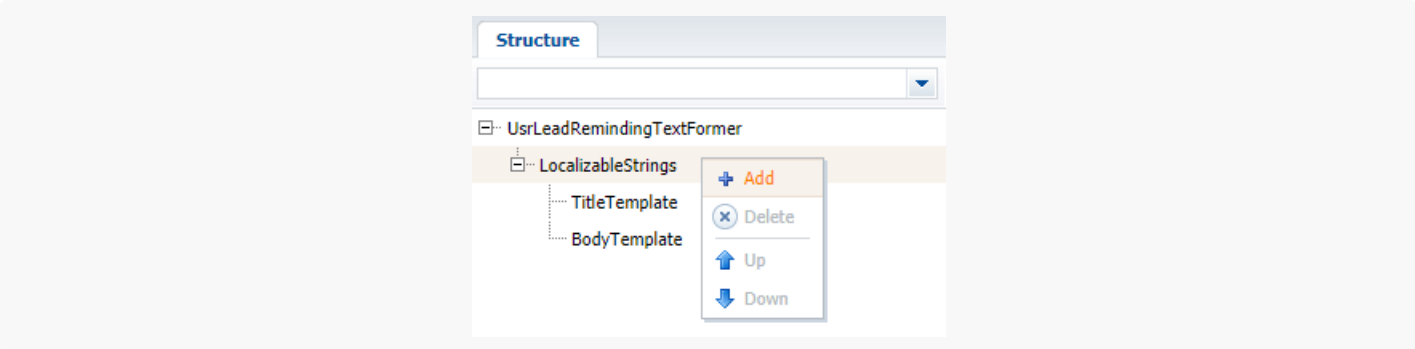


Табл. 1. — Свойства локализуемых строк

[Название] ([Name])	[Значение] ([Value])
TitleTemplate	Необходимо актуализировать продажу (You need to update the sale)
BodyTemplate	Лид {0} требует актуализации информации по продаже (Lead {0} requires update of sales information)

3. На вкладку [*Исходный код*] ([*Source code*]) добавьте реализацию класса для формирования текста напоминания и всплывающего окна:

```
namespace Terrasoft.Configuration
```

```

{
    using System.Collections.Generic;
    using Terrasoft.Common;
    using Terrasoft.Core;

    public class UsrLeadRemindingTextFormer : IRemindingTextFormer
    {
        private const string ClassName = nameof(UsrLeadRemindingTextFormer);
        protected readonly UserConnection UserConnection;

        public UsrLeadRemindingTextFormer(UserConnection userConnection) {
            UserConnection = userConnection;
        }
        // Формирует текст напоминания из коллекции входящих параметров и локализуемой строки Body
        public string GetBody(IDictionary<string, object> formParameters) {
            formParameters.CheckArgumentNull("formParameters");
            var bodyTemplate = UserConnection.GetLocalizableString(ClassName, "BodyTemplate");
            var leadName = (string)formParameters["LeadName"];
            var body = string.Format(bodyTemplate, leadName);
            return body;
        }
        // Формирует заголовок напоминания из имени класса и локализуемой строки TitleTemplate.
        public string GetTitle(IDictionary<string, object> formParameters) {
            return UserConnection.GetLocalizableString(ClassName, "TitleTemplate");
        }
    }
}

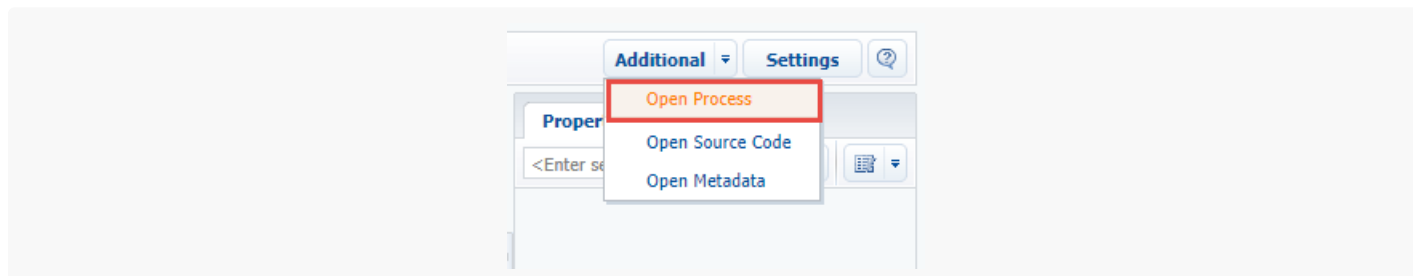
```

4. Сохраните и опубликуйте схему.

2. Заместить объект [Лид] и задать в нем логику отправки напоминания

1. Создайте замещающую схему объекта [Лид] (см. раздел "Создание замещающей схемы объекта" статьи "[Создать схему объекта](#)").
2. Используя команду [Дополнительно] ([Additional]) — [Открыть процесс] ([Open process]), перейдите в дизайнер встроенного процесса замещающего объекта [Лид] (рис. 3).

Рис. 3. — Команда перехода в дизайнер встроенного процесса



3. Используя контекстное меню вкладки [Структура] ([Structure]), добавьте параметр процесса `GenerateReminding` (рис. 4) со следующими свойствами (рис. 5):

- [Заголовок] ([Title]) — "Сгенерировать напоминание" ("Generate Reminding").
- [Название] ([Name]) — "GenerateReminding".
- [Тип данных] ([Data type]) — "Boolean".

Рис. 4. — Добавление параметра процесса

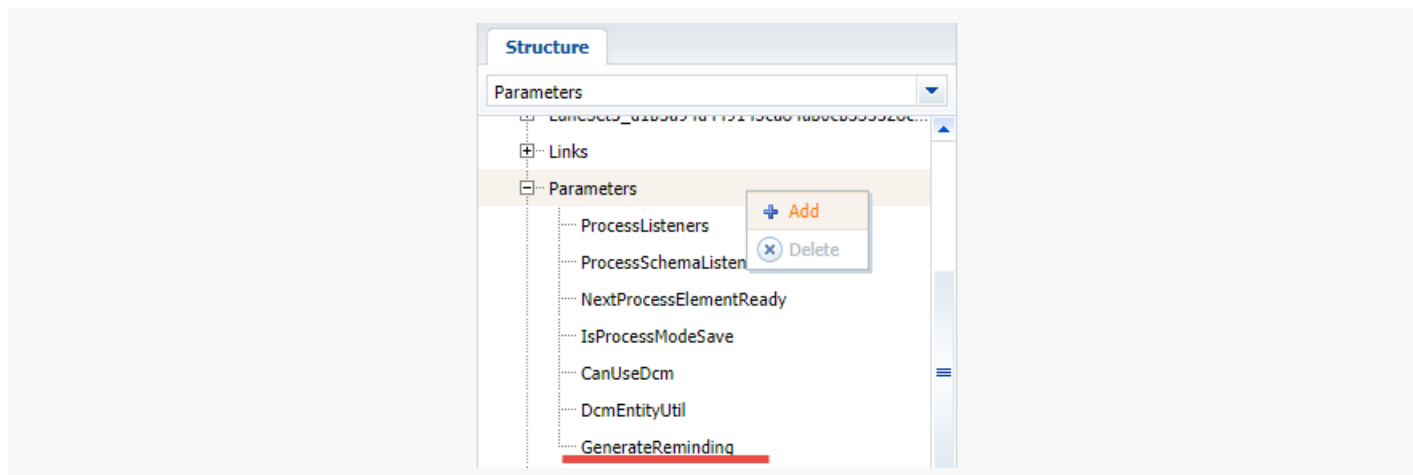
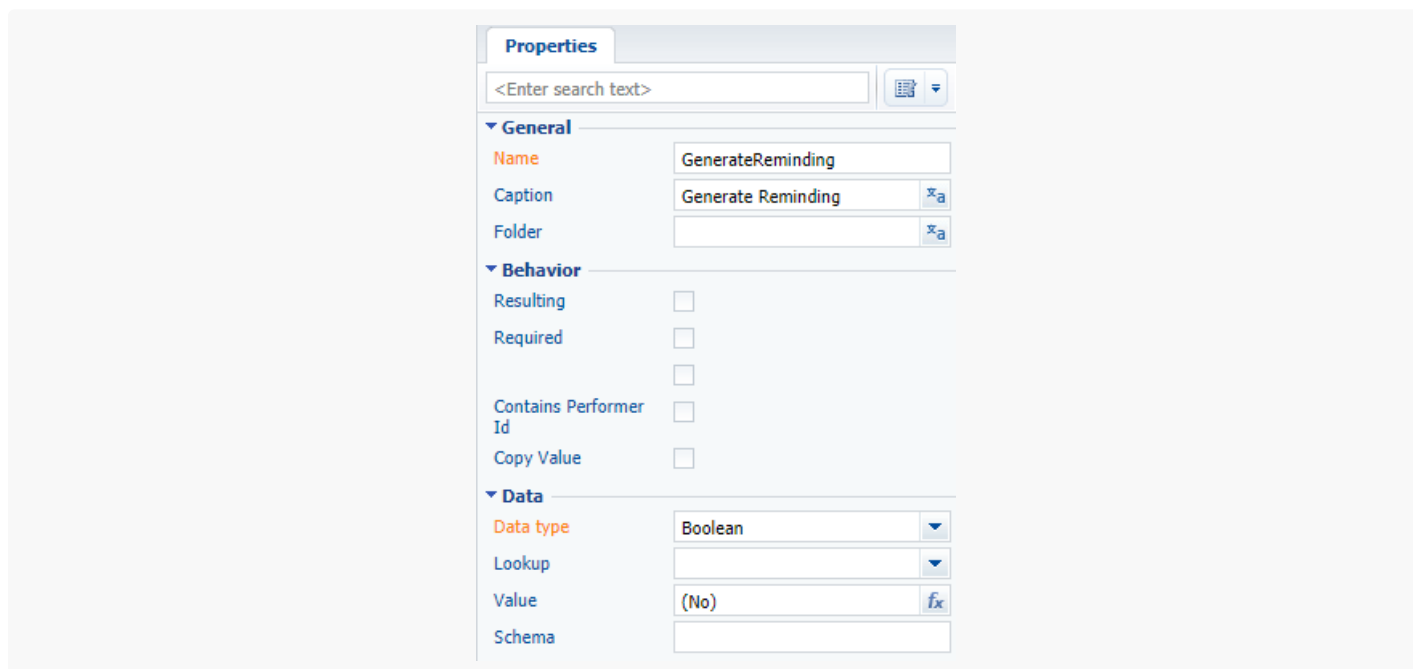


Рис. 5. — Свойства параметра процесса



4. На вкладке [Структура] ([Structure]), в группе [Methods] выберите метод `LeadSavingMethod()`, который вызывается перед сохранением объекта (рис. 6). Укажите признак [Переопределен] ([Override]) и добавьте следующий исходный код в тело метода:

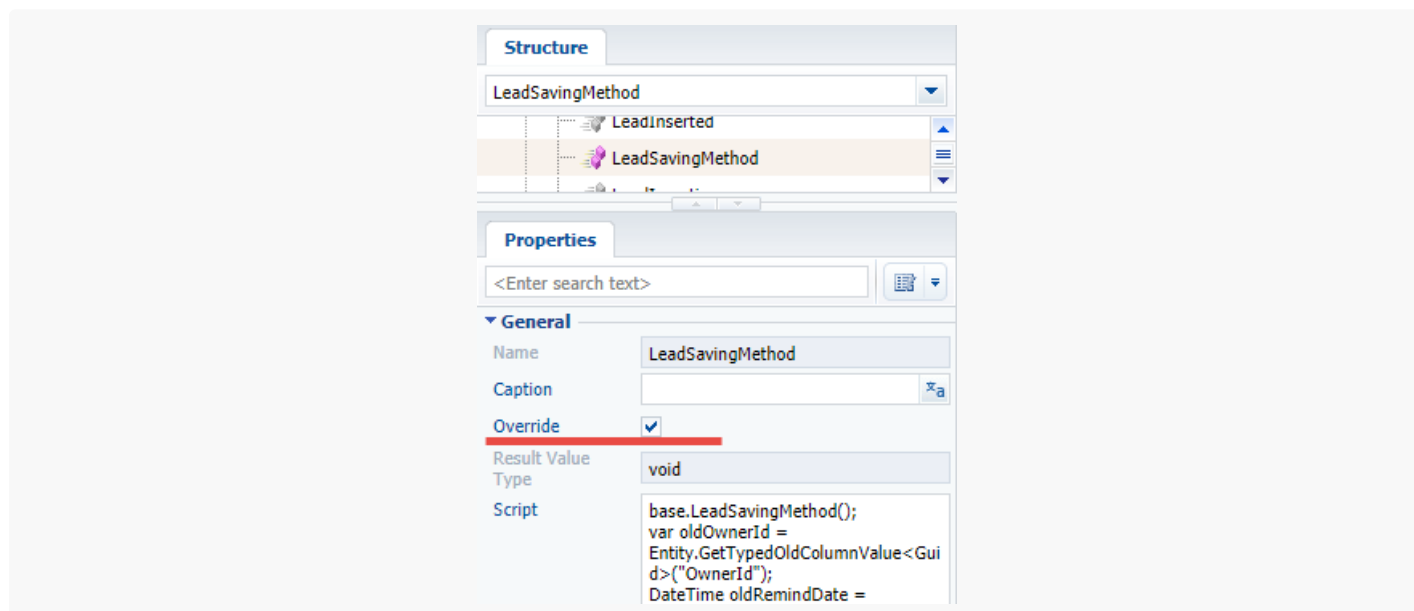
```
// Вызов базовой реализации метода.
base.LeadSavingMethod();
```

```
// Получение идентификатора ответственного.
var oldOwnerId = Entity.GetTypedOldColumnValue<Guid>("OwnerId");
// Получение даты следующей актуализации.
DateTime oldRemindDate = Entity.GetTypedOldColumnValue<DateTime>("NextActualizationDate");
// Сравнение старого значения идентификатора ответственного и нового.
bool ownerChanged = !Entity.OwnerId.Equals(oldOwnerId);
// Сравнение старого значения даты актуализации и нового.
bool remindDateChanged = !Entity.NextActualizationDate.Equals(oldRemindDate);
// Параметру процесса GenerateReminding присвоить true, если были изменены или ответственный,
// или дата следующей актуализации.
GenerateReminding = ownerChanged || remindDateChanged;
```

На заметку.

Для отображения редактора исходного кода тела метода дважды кликните по названию метода на вкладке [Структура].

Рис. 6. — Свойства метода LeadSavingMethod



5. На вкладке [Структура] ([Structure]), в группе [Methods] выберите метод `LeadSavedMethod()`, который вызывается после сохранения объекта (рис. 7). Укажите признак [Переопределен] ([Override]) и добавьте следующий исходный код в тело метода:

```
base.LeadSaved();
// Проверки необходимости генерации напоминания.
if (!GenerateReminding) {
    return;
}
DateTime remindTime = Entity.NextActualizationDate;
if (Entity.OwnerId.Equals(Guid.Empty) || remindTime.Equals(default(DateTime))) {
```

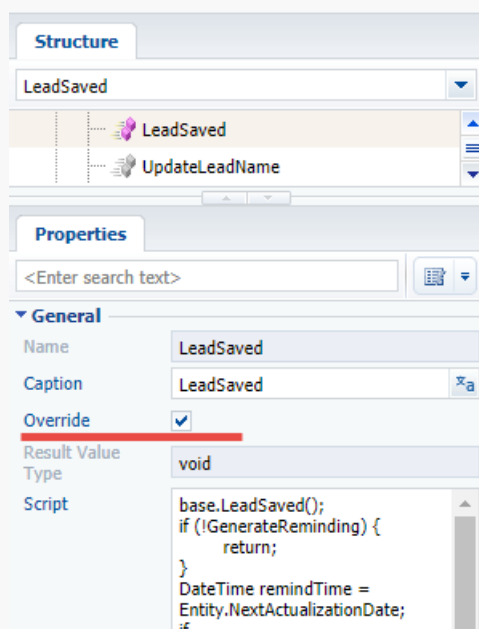


```


        return;
    }
    // Создание экземпляра класса UserLeadRemindingTextFormer.
    IRemindingTextFormer textFormer =
        ClassFactory.Get<UserLeadRemindingTextFormer>(new ConstructorArgument("userConnection", UserC
    // Получение названия лида.
    string leadName = Entity.LeadName;
    // Получение текста напоминания.
    string subjectCaption = textFormer.GetBody(new Dictionary<string, object> {
        {"LeadName", leadName}
    });
    // Получение заголовка напоминания.
    string popupTitle = textFormer.GetTitle(null);
    // Конфигурирование напоминания.
    var remindingConfig = new RemindingConfig(Entity);
    // Автор сообщения – текущий контакт.
    remindingConfig.AuthorId = UserConnection.CurrentUser.ContactId;
    // Целевой получатель – ответственный лида.
    remindingConfig.ContactId = Entity.OwnerId;
    // Тип – напоминание.
    remindingConfig.NotificationTypeId = RemindingConsts.NotificationTypeRemindingId;
    // Дата отправки напоминания – дата актуализации продажи в лиде.
    remindingConfig.RemindTime = remindTime;
    // Текст напоминания.
    remindingConfig.Description = subjectCaption;
    // Заголовок напоминания.
    remindingConfig.PopupTitle = popupTitle;
    // Создание утилитного класса напоминания.
    var remindingUtilities = ClassFactory.Get<RemindingUtilities>();
    // Создание напоминания.
    remindingUtilities.CreateReminding(UserConnection, remindingConfig);

```

Рис. 7. — Свойства метода LeadSavedMethod



На заметку.

Чтобы напоминания отображались на вкладке системных уведомлений , необходимо в коде метода LeadSavedMethod вместо

```
remindingConfig.NotificationTypeId = RemindingConsts.NotificationTypeRemindingId; указать
remindingConfig.NotificationTypeId = RemindingConsts.NotificationTypeNotificationId;
```

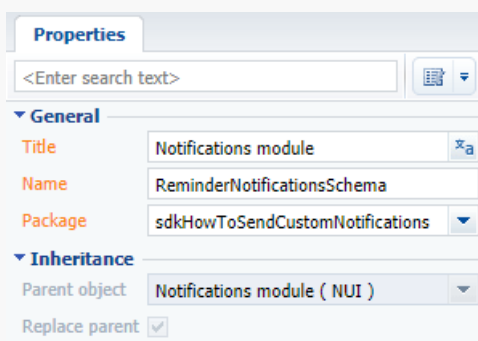
6. Сохраните и опубликуйте схему встроенного процесса объекта [Лид]. Затем опубликуйте схему объекта [Лид].

3. Заместить схему вкладки напоминаний ReminderNotificationsSchema

1. Чтобы отображались напоминания по требуемому объекту, создайте в [пользовательском пакете](#) замещающую схему `ReminderNotificationsSchema` и добавьте в нее необходимую программную логику. Как создать замещающую клиентскую схему описано в статье ["Создать клиентскую схему"](#). Свойства замещающей схемы (рис. 8):

- [Заголовок] ([Title]) — "Модуль уведомлений" ("Notifications module").
- [Название] ([Name]) — "ReminderNotificationsSchema".

Рис. 8. — Свойства схемы ReminderNotificationsSchema



2. На вкладку [*Исходный код*] ([*Source code*]) схемы добавьте следующий исходный код:

```
define("ReminderNotificationsSchema", ["ReminderNotificationsSchemaResources"],
function() {
    return {
        entitySchemaName: "Reminding",
        methods: {
            // Определяет, относится ли напоминание к лиду.
            getIsLeadNotification: function() {
                return this.get("SchemaName") === "Lead";
            },
            // Возвращает заголовок напоминания.
            getNotificationSubjectCaption: function() {
                var caption = this.get("Description");
                return caption;
            }
        },
        // Массив модификаций модели представления.
        diff: [
            // Основной контейнер напоминания.
            {
                "operation": "insert",
                "name": "NotificationleadItemContainer",
                "parentName": "Notification",
                "propertyName": "items",
                "values": {
                    "itemType": Terrasoft.ViewItemType.CONTAINER,
                    "wrapClass": [
                        "reminder-notification-item-container"
                    ],
                    // Отображается только для лидов.
                    "visible": {"bindTo": "getIsLeadNotification"},
                    "items": []
                }
            },
            // Контейнер для заголовка.
            {
```

```

        "operation": "insert",
        "name": "NotificationItemleadTopContainer",
        "parentName": "NotificationleadItemContainer",
        "propertyName": "items",
        "values": {
            "itemType": Terrasoft.ViewItemType.CONTAINER,
            "wrapClass": ["reminder-notification-item-top-container"],
            "items": []
        }
    },
    // Изображение.
    {
        "operation": "insert",
        "name": "NotificationleadImage",
        "parentName": "NotificationItemleadTopContainer",
        "propertyName": "items",
        "values": {
            "itemType": Terrasoft.ViewItemType.BUTTON,
            "className": "Terrasoft.ImageView",
            "imageSrc": {"bindTo": "getNotificationImage"},
            "classes": {"wrapClass": ["reminder-notification-icon-class"]}
        }
    },
    // Отображение даты.
    {
        "operation": "insert",
        "name": "NotificationDate",
        "parentName": "NotificationItemleadTopContainer",
        "propertyName": "items",
        "values": {
            "itemType": Terrasoft.ViewItemType.LABEL,
            "caption": {"bindTo": "getNotificationDate"},
            "classes": {"labelClass": ["subject-text-labelClass"]}
        }
    },
    // Отображение текста напоминания.
    {
        "operation": "insert",
        "name": "NotificationleadSubject",
        "parentName": "NotificationItemleadTopContainer",
        "propertyName": "items",
        "values": {
            "itemType": Terrasoft.ViewItemType.LABEL,
            "caption": {"bindTo": "getNotificationSubjectCaption"},
            "click": {"bindTo": "onNotificationSubjectClick"},
            "classes": {"labelClass": ["subject-text-labelClass", "label-link", "lat
        }
    }
}
]

```

```
};
});
```

3. Сохраните схему.

В результате выполнения примера для лидов, у которых указаны ответственный и дата следующей актуализации (рис. 9), будут приходить напоминания о необходимости актуализации (рис. 10).

Рис. 9. — Лид с указанными ответственным и датой следующей актуализации

The screenshot shows the 'Additional service' lead record. The sidebar on the left contains the following information:

- Customer need***: Additional service
- Registration method**: Added manually
- Budget**: 0.00
- Created on**: 3/1/2018 12:24 PM
- Owner**: Supervisor

The main panel displays a process flow with stages: Qualification (active), Nurturing, Handoff to sales, Awaiting sale, and Satisfied. Below the flow, it shows 'NEXT STEPS (0)' with icons for phone, email, chat, and flag. A tabbed interface at the bottom includes 'LEAD INFO', 'CUSTOMER NEED DETAILS', 'LEAD ENGAGEMENT', 'TIMELINE', and 'OPPORTUNITY INFO' (selected). The 'Opportunity information' section shows:

- Budget**: 0.00
- Next actualization date**: 3/2/2018 12:00 PM
- Opportunity/Order**: (empty field)
- Deal owner**: (empty field)

Рис. 10. — Напоминание об актуализации лида

The screenshot shows a notification reminder. At the top, there is a row of icons: a clock with a red '1' badge, a speech bubble, a checkmark, a gift, and an information icon. Below these are buttons for 'Postpone all' and 'Cancel all'. The main notification text reads: 'Today at 12:00 PM Lead Additional service requires update of sales information'. To the right of the notification is a vertical toolbar with icons for settings, help, phone, email, chat, a bell with a red '1' badge, and a document icon.