

сократить время выполнения задач за счет автоматизации процессов.

Система учета задолженности по банковским картам должна взаимодействовать с другими системами банка. Для этих целей банки разрабатывают системы самостоятельно или покупают комплектные готовые решения для работы банка.

Примеры внешних систем, представленных на рынке Республики Беларусь:

1. EPAM.Debt Collection – это гибко настраиваемая система, предназначенная для розничных банков;
2. Системные технологии Collection – система для предупреждения возникновения задолженности и управления процессами взыскания.

Список использованных источников:

1. Налогообложение, учет и отчетность в коммерческом банке //Методический журнал «Бухгалтерский учет просроченной задолженности физических лиц». – 2011. - №4 – С. 5-10.

2. Мирошниченко, Ю. В. Работа с просроченной задолженностью // InfoBank.by [Электронный ресурс]. - Режим доступа: <http://infobank.by/rabota-s-prosrochennoj-zadolzhennostju.html>. – Дата доступа: 23.11.2017.

3. Лопатин, Ю. В. Аналитический отчет «ИССЛЕДОВАНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В БАНКОВСКОМ СЕКТОРЕ РЕСПУБЛИКИ БЕЛАРУСЬ» // Научно-технологическая ассоциация "Инфопарк" by [Электронный ресурс]. - Режим доступа: <http://infopark.by/sites/default/files/old/docs/7660.pdf>. – Дата доступа: 24.11.2017.

## РОБОТЫ В CRM БИТРИКС24. REST ДЛЯ CRM-ТРИГГЕРОВ

*Институт информационных технологий БГУИР,  
г. Минск, Республика Беларусь*

*Литовко Д.А.*

*Бакунова О.М. – ст. преподаватель каф. ИСиТ, м.т.н.*

*Бакунов А.М. – ст. преподаватель каф. ИСиТ, м.т.н.*

*Калетня И.Л. – ассистент каф. ИСиТ, м.т.н.*

*Хмелевская А.Л. – ассистент каф. ПЭ*

В современном обществе большинство компаний начинают искать различные способы для автоматизации ведения бизнеса. Для этого были созданы различные CRM, которые помогают вести бизнес и упрощают большую часть бизнес-процессов, экономя этим время и финансы организации. В докладе представлен пример использования в бизнесе CRM Битрикс24 [1].

В CRM Битрикс24 существует обширный REST API, который позволяет добавлять функционал как в облачный, так и в коробочный Битрикс24, а также создавать всевозможные интеграции с внешними системами. Использование методов REST позволяет создавать и модифицировать объекты CRM, задачи, публикации, сделки и т.д.

Работа с клиентами и продажами состоит из множества рутинных операций - планирование встреч, работа с почтой и звонками, постановка задач и контроль их выполнения. CRM Битрикс24 предоставляет огромное количество различных инструментов, которые автоматизируют эти операции до максимально возможного уровня. Одним из таких инструментов являются так называемые роботы.

Пользователи могут создавать и настраивать роботов, которые будут отсылать различные нотификации, ставить задачи при поступлении нового лица, выполнять почтовые рассылки, управлять статусами сделок, планировать встречи и т.д. Роботы позволяют решать базовые задачи автоматизации бизнеса, используя простые линейные алгоритмы. За счет своей линейности являются простым и эффективным инструментом. Для достижения максимальной автоматизации в связке с роботами используются триггеры.

Для использования триггеров необходимо их создать, либо использовать уже существующие (CRM содержит набор базовых триггеров). Создаются триггеры с помощью приложений Битрикс24 [2]. Приложения Битрикс24 делятся на 3 типа:

- Внешние серверные приложения (приложения используют только API, и никак не интегрируются в интерфейс Битрикс24. Доступ для использования API получается посредством протокола авторизации OAuth 2.0);

- Серверные приложения (размещаются на сторонних серверах);

- Статические приложения (размещаются в облаке 1С-Битрикс).

Непосредственно такие приложения и содержат в себе описание событий и линейные алгоритмы триггеров. При добавлении нового триггера, необходимо описать всю цепочку действий, которые данный триггер будет выполнять. После добавления триггера, необходимо описать событие, при наступлении которого будет вызван данный триггер.

Когда все триггеры и события будут описаны необходимо установить приложение в CRM. После установки приложения появится возможность устанавливать триггеры, на различные события (рисунок 1).

Триггеры можно использовать для разных целей и вот одна из возможных:

В CRM, на определенном этапе бизнес-процесса, роботы выполняют почтовую рассылку для определенного списка лиц. В письмо попадает коммерческое предложение для конкретного лица. Делается это для привлечения лица и перевода его в статус клиента. Вместе с предложением отправляется уникальная ссылка на нужную страницу, где можно подробнее ознакомиться с полученным предложением. В ссылку встраиваются данные лица (далее клиент):

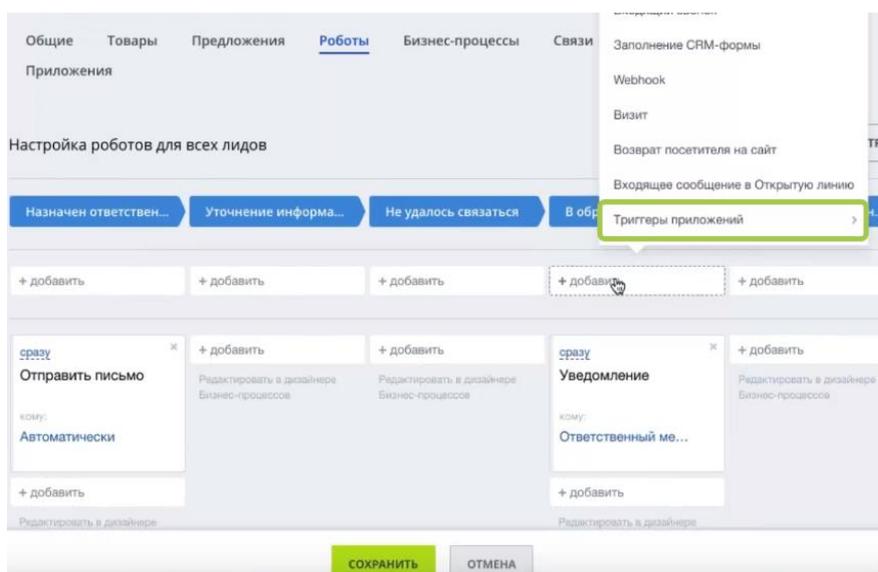


Рисунок 1 – Пример добавления триггера в CRM

- Тип клиента (группа, к которой принадлежит клиента);
- Уникальный идентификатор клиента.

В том случае, если клиент переходит по указанной ссылке, он попадает на страницу организации. В этот момент страница получает данные этого клиента. Эти данные обрабатываются и с помощью ajax-запроса отсылаются во «внешнее приложение». Это приложение в последствии и будет взаимодействовать с данными и использовать различные REST запросы для манипулирования CRM.

Если приложение получило все необходимые данные, тогда через REST API выполняется запрос в CRM. Это и есть событие. Чтобы событие сработало правильно, необходимо передать минимальный набор параметров:

- Идентификатор триггера (символьный код);
- Данные клиента.

Пример события показан на рисунке 2.

CRM обрабатывает событие и вызывает указанный триггер. В свою очередь триггер начинает выполнять свой алгоритм. В данном случае триггер переводит клиента на другой этап бизнес-процесса и создает нотификацию о том, что конкретный клиент, в данный момент времени, просматривает страницу с коммерческим предложением. После этого менеджер, который отвечает за данного клиента получает нотификацию о действиях клиента. Менеджер связывается с этим клиентом и начинает предлагать свои услуги, либо свою помощь.

```
$result = restCommand('crm.automation.trigger.execute', //событие
    array(
        'CODE' => 'символьный_код_триггера',
        'OWNER_TYPE_ID' => intval($_REQUEST['TYPE']), // тип клиента в CRM
        'OWNER_ID' => intval($_REQUEST['ID']) // идентификатор клиента в CRM
    ),
    array('domain' => $_REQUEST['DOMAIN'], 'access_token' => $_REQUEST['AUTH_ID']) //конфигурации приложения
);
```

Рисунок 2 – Пример отправки события с помощью REST запроса

Этот простой алгоритм применения триггеров представлен на рисунке 3.

Это далеко не единственно возможный способ применения роботов, триггеров и REST запросов. Эти инструменты являются огромной средой для творческой деятельности разработчиков.

Для подобных целей так же подходят Вебхуки, принцип работы схож, однако технологии разные, а соответственно и способы достижения поставленной цели будут другими.

Работа с REST не простая задача, которая требует должной квалификации разработчика. Однако при умелом использовании REST, при грамотной настройке бизнес-процессов, роботов, событий, триггеров и сторонних приложений позволяет максимально упростить работу менеджера при ведении клиента по воронке продаж, от статуса лица, до заключения с ним сделки.



Рисунок 3 – Пример применения триггеров

Список используемых источников

1. CRM Битрикс24. [Электронный ресурс]. – Режим доступа: <https://www.bitrix24.by/>. – Дата доступа: 04.03.2018.
2. Создание приложений Битрикс24 [Электронный ресурс]. — Режим доступа: <https://www.bitrix24.ru/apps/dev.php> – Дата доступа: 04.03.2018.

## ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ОБУЧЕНИЯ ОСНОВАМ КРИПТОГРАФИИ

*Институт информационных технологий БГУИР,  
г. Минск, Республика Беларусь*

*Лупач В.Д.*

*Калитеня И.Л. – ассистент каф ИСиТ, м.т.н.*

Сегодня едва ли удастся найти компанию, на компьютерах которой не обрабатывалась и не хранилась бы важная коммерческая информация. Полноценная система ее защиты имеет исключительное значение и без криптографии здесь никак не обойтись. Применительно к компаниям шифрование обычно используется в трех случаях, а именно в целях обеспечения безопасности хранения данных, защиты информации при передаче через открытые каналы связи и по локальной сети, наконец, для цифровых подписей. Все представленные задачи решаются разными средствами с применением различных криптографических технологий.

В современных корпоративных сетях вся важная информация сосредоточена на одном или нескольких серверах. Такой подход по сравнению с хранением данных на компьютерах конечных пользователей имеет множество преимуществ. Первостепенной задачей здесь является обеспечение надежного шифрования документов именно на сервере, но при этом система безопасности не должна серьезно осложнять работу сотрудников с информацией, которая необходима для исполнения служебных обязанностей.

Знание криптографических методов важно для развития и профессионального роста программистов в современном мире. Программисты с подобными знаниями смогут профессионально заниматься защитой корпоративной информации.

В дипломном проекте ставилась задача создать программное средство, которое могло бы обучать практическим основам криптографии и проверять знания учащихся. Ставилась задача создать программное средство, которое помогало бы в процессе обучения, а также в котором учащийся смог бы самостоятельно изучить пропущенные темы. Процесс обучения, как и процесс проверки знаний, проходит в несколько этапов. В режиме обучения, при неправильных действиях учащегося, подсказки в каждом этапе помогут найти и исправить свою ошибку. При проверке знаний, учащийся будет иметь возможность проверить свой ответ и продемонстрировать процесс решения.

Программное средство имеет следующие функции:

- авторизация учащегося;
- выбор режима администратора;
- выбор криптографического метода;
- реализация двух методов: обучения и проверки знаний;
- реализация возможности ввода любых текстов для зашифровки и любых возможных ключей;
- помощь и подсказки при обучении;
- проверка знаний по какому-либо криптографическому методу;
- сохранение оценки пользователя в базе данных;
- сохранение в базе данных времени, затраченного на выполнение учащимся задания;
- просмотр оценок учащихся администратором.