

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

«Crosstech Smart Assets (CTSA)»

**ФУНКЦИОНАЛЬНЫЕ ХАРАКТЕРИСТИКИ,
УСТАНОВКА И ЭКСПЛУАТАЦИЯ**

© ООО «Кросстех Солюшнс Групп»

2022

1. ОГЛАВЛЕНИЕ

| | |
|---|-----------|
| 2. ОПИСАНИЕ СИСТЕМЫ..... | 5 |
| 3. СОСТАВ СИСТЕМЫ. ОБЩАЯ СХЕМА РЕШЕНИЯ. | 6 |
| 4. ЦЕЛИ И ЗАДАЧИ | 7 |
| 5. РАЗВЁРТЫВАНИЕ..... | 8 |
| 6. ЗАПУСК СИСТЕМЫ | 10 |
| 7. ЛИЦЕНЗИРОВАНИЕ СИСТЕМЫ..... | 11 |
| 8. ТРЕБОВАНИЯ К АППАРАТНОМУ ОБЕСПЕЧЕНИЮ..... | 14 |
| 9. ЭКСПЛУАТАЦИЯ СИСТЕМЫ..... | 18 |
| 9.1 Жизненный цикл Актива | 18 |
| События жизненного цикла | 19 |
| Информационные системы | 20 |
| Основные средства/комплекты..... | 20 |
| 9.2 Мастер приемки активов | 22 |
| Мастер перемещения активов | 23 |
| Проведение инвентаризации..... | 23 |
| Разделение партии активов | 26 |
| 9.3 Управление соглашениями..... | 26 |
| Договоры..... | 26 |
| Позиции договора | 27 |
| Дополнительные соглашения | 27 |
| Счета..... | 28 |
| Заявки на оплату | 28 |
| Документы | 28 |
| События договора | 29 |
| 9.4 Управление финансами | 29 |
| МВЗ | 29 |
| Статьи расходов | 29 |
| Расходы | 30 |
| Валюты и курсы | 31 |
| Ставки НДС | 32 |
| 9.5 Управление делами..... | 33 |

| | |
|---|-----------|
| Статусы дел | 34 |
| Заявки | 35 |
| Запросы на обслуживание..... | 37 |
| Наряды на работу | 39 |
| Эскалация дел..... | 40 |
| Эскалация заявок..... | 41 |
| Эскалация запросов | 41 |
| 9.6 Организационная структура | 41 |
| Подразделения..... | 41 |
| Сотрудники и пользователи..... | 42 |
| Рабочие группы | 43 |
| 9.7 Управление активами ПО (SAM) | 43 |
| Приложения..... | 43 |
| Лицензии..... | 44 |
| Типы лицензий | 45 |
| Правообладатели дискаверинга..... | 46 |
| Наименования дискаверинга | 47 |
| Нормализация данных | 48 |
| Нормализация правообладателей..... | 48 |
| Нормализация наименований | 48 |
| Полномочия | 49 |
| Управление полномочиями..... | 49 |
| Метрики SAM..... | 51 |
| Счётчики SAM | 51 |
| 9.8 Справочники..... | 54 |
| Календари рабочих дней | 55 |
| Контрагенты | 56 |
| Местоположения | 56 |
| Правообладатели..... | 57 |
| Наименования..... | 58 |
| Наименования приложений | 59 |
| Наименования ИТ-оборудования | 59 |

| | |
|--|-----------|
| Наименования лицензий | 60 |
| Категории..... | 60 |
| Проекты..... | 60 |
| Склады..... | 61 |
| Единицы измерения | 62 |
| 10. ИНТЕГРАЦИИ..... | 63 |
| LDAP | 63 |
| Интеграция с Active Directory с использованием Jespa | 63 |
| OCS inventory NG..... | 64 |
| Сайт-менеджер | 64 |
| Сайт ЦБ РФ..... | 64 |
| Битрикс24..... | 64 |
| Email | 64 |

2. ОПИСАНИЕ СИСТЕМЫ

Система управления активами CrossTech Smart Assets (далее — Система) предназначена для управления процессами физического и финансового учета активов, входящих в состав ИТ-решений любой сложности с целью повышения прозрачности учёта активов, сокращения финансовых затрат на лицензирование программного обеспечения и повышения эффективности работы ИТАМ-процедур предприятия.

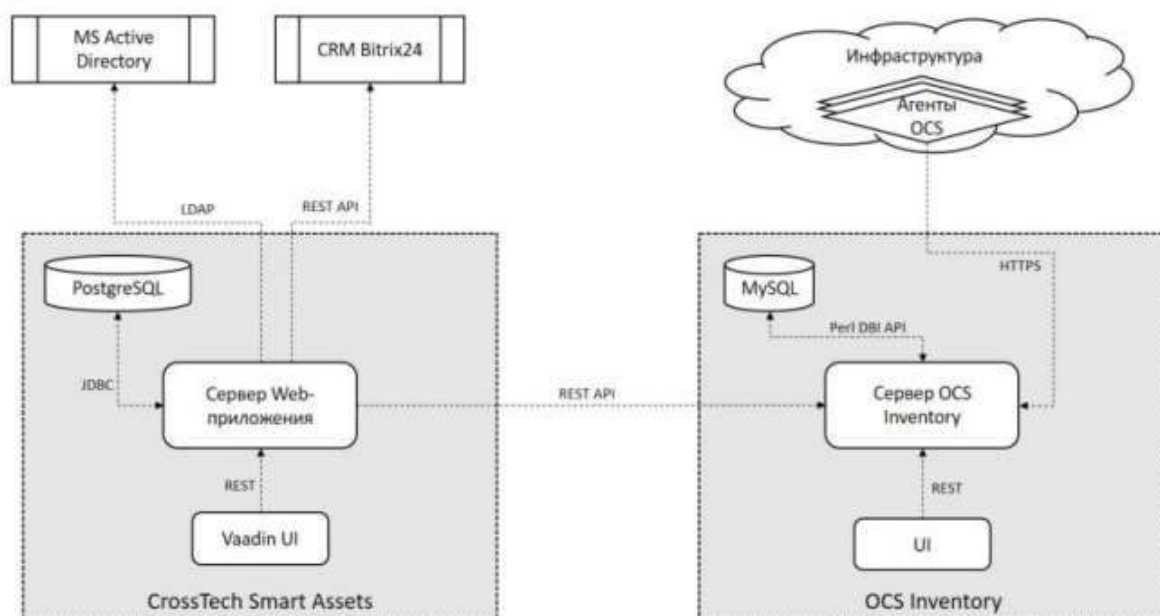
Система обеспечивает получение актуальной информации об активах компании и отражает их текущее состояние.

Система основана на высокоуровневой Java-платформе для создания корпоративных информационных систем CUBA Platform с открытым исходным кодом (Open Source) версии 7.2.

3. СОСТАВ СИСТЕМЫ. ОБЩАЯ СХЕМА РЕШЕНИЯ.

Общая схема решения

Система имеет следующую архитектуру:



Решение состоит из двух приложений, работающих независимо:

OCS Inventory представляет из себя клиент-серверное приложение, имеющее веб-интерфейс. Собирает информацию с помощью агентов — устанавливаемых на конечные устройства программ (Windows, UNIX и MacOS), осуществляющих периодический сбор информации по устройствам. Собранную информацию хранит в СУБД MySQL.

CrossTech Smart Assets является клиент-серверным приложением, имеющим веб-интерфейс, а также предоставляющим множество возможностей (формирование отчетности, собственный REST API, реализацию BPM и др.). Обеспечивает полный цикл процесса управления активами. В качестве БД используется PostgreSQL (есть возможность использовать другие СУБД). По умолчанию включены интеграции с Active Directory, CRM Bitrix24 (существует возможность включать интеграции с другими системами). Система состоит из Middleware-слоя и UI (Vaadin).

4. ЦЕЛИ И ЗАДАЧИ

1. Основная цель Системы – автоматизация процессов физического и финансового учета материальных и нематериальных активов.

Система решает следующие задачи:

- Создание единого информационного пространства в части учёта активов компании;
- Автоматизация управления жизненным циклом материальных и нематериальных активов;
- Управление активами программного обеспечения в части сопоставления обнаруженного ПО с закупленными лицензиями;
- Ведение реестра договоров и документов, связанных с активами;
- Автоматизация процедур приёма и постановки активов на учёт, перемещения активов, учёта договоров и поставок, классификации и категоризации активов, управления делами (заявками, запросами и нарядами на работу) в части процедуры управления активами, отслеживания финансовой составляющей активов, проведения инвентаризации;
- Построение статистических графиков, информационных панелей и отчётов для получения аналитической и управленческой отчётности.

2. Выполнение требований законодательства, в том числе:

— Стандарт Банка России СТО БР ИББС-1.0-2014:

Должны проводиться процедуры учета и классификации информационных активов организации.

5. РАЗВЁРТЫВАНИЕ

В данном разделе описаны основные этапы развёртывания Системы, а также требования к подготовке и настройке окружения для стабильной работы.

Подготовка инфраструктуры

Перед развёртыванием Системы необходимо определить количество одновременно работающих пользователей и развернуть конфигурацию серверов в соответствии с требованиями к аппаратному обеспечению.

Необходимо организовать все необходимые сетевые доступы и предоставить требуемые для работы системы права служебным учётным записям.

Установка среды разработки

Для эффективной поддержки и возможности внесения доработок/корректировок в существующую систему необходимо развернуть исходный код проекта в CUBA Studio.

Следующие инструменты упрощают разработку на фреймворке CUBA:

CUBA Studio — это интегрированная среда разработки (IDE), построенная на платформе IntelliJ IDEA и содержащая все необходимое для работы с проектами на CUBA. Ее можно установить либо как отдельное приложение для вашей операционной системы, либо в виде плагина для IntelliJ IDEA (Community or Ultimate).

CUBA CLI - инструмент командной строки, предоставляющий базовую функциональность для создания проектов и их элементов: сущностей, экранов, сервисов и т.д. Этот инструмент позволяет использовать для разработки приложений на CUBA любую Java IDE.

С инструкцией по установке CUBA Studio можно ознакомиться в официальной документации по ссылке.

Системные требования для CUBA Studio:

- 64-битная операционная система - Windows, Linux или macOS;
- Оперативная память - 8 ГБ для ведения разработки в CUBA Studio;
- Свободное место на жестком диске - 10 ГБ.

Установка CUBA Studio возможна как на сервер, так и на локальное рабочее место.

После внесения изменений в исходный код проекта, указания необходимых параметров и свойств необходимо собрать проект заново, например в:

- UberJAR (инструкция по ссылке);
- WAR (инструкция по ссылке).

После чего будет необходимо развернуть приложение на сервере приложений.

Развёртывание Системы

Собранное приложение Системы может представлять собой как WAR-архив, так и исполняемый UberJAR-файл.

В обоих случаях на сервере приложений должна быть установлена Java SE Development Kit (JDK) версии 11. Необходимо установить OpenJDK 11 и проверить его работоспособность, выполнив в консоли команду:

```
java -version
```

В ответ должно быть выведено сообщение с номером версии Java, например:

```
openjdk version "11" 2018-09-25
OpenJDK Runtime Environment 18.9 (build 11+28)
OpenJDK 64-Bit Server VM 18.9 (build 11+28, mixed mode)
```

Если версия java не указана или отличается от 11, то необходимо установить путь к корневому каталогу JDK в переменной окружения JAVA_HOME, например C:\Java\jdk11.

Для Windows это можно сделать, открыв Компьютер → Свойства системы → Дополнительные параметры системы → Дополнительно → Переменные среды, и задав значение переменной в списке.

Системные переменные

Для macOS рекомендуется установить JDK в каталог /Library/Java/JavaVirtualMachines, например /Library/Java/JavaVirtualMachines/jdk11, и задать JAVA_HOME в ~/.bash_profile

следующей командой:

```
export JAVA_HOME="$(/usr/libexec/java_home -v 11)"
```

Если для соединения с интернетом используется прокси-сервер, в JVM, исполняющие инструменты разработки и Gradle, необходимо передавать специальные системные свойства Java, описанные в документе (см. свойства для протоколов HTTP и HTTPS). Рекомендуется установить нужные свойства в переменной окружения JAVA_OPTS для всей операционной системы.

6. ЗАПУСК СИСТЕМЫ

Перед запуском Системы необходимо подложить активную лицензию, полученную для вашей организации. В зависимости от типа развёртывания лицензию необходимо разместить в соответствующей директории. Подробно про данный процесс описано в разделе Установка лицензии на сервер приложений.

После установки OpenJDK 11 и размещения в нужной директории лицензии на сервер приложений для запуска приложения необходимо либо запустить UberJar (инструкция здесь), либо, например, развернуть WAR-архив на Tomcat для Windows (инструкция здесь). В официальной документации CUBA Platform приведены все возможные варианты развёртывания в зависимости от требуемой конфигурации.

В качестве базы данных по умолчанию выступает PostgreSQL. При развёртывании приложения на сервер необходимо указать настройки подключения к базе данных PostgreSQL, после чего Система сама создаст пустую БД с необходимым для работы стартовым набором сущностей.

Веб-интерфейс приложения, поддерживает все популярные современные браузеры, в том числе: Google Chrome, Mozilla Firefox, Safari, Opera 15+, Internet Explorer 11, Microsoft Edge.

Установка мобильного приложения CTSA Inventory

Приложение для проведения инвентаризации CTSA Inventory входит в поставку с сервером Crosstech Smart Assets и является частью Системы. Возможность использования мобильного приложения определяется наличием на него лицензии.

Для Android:

1. Установить приложение `ctsa_inventory.apk` на Android-смартфон;
2. Установить на телефон VPN для доступа к внутренним ресурсам организации, в частности, к серверу Crosstech Smart Assets (например, GlobalProtect из GooglePlay) и настроить подключение;
3. При первом запуске указать в настройках параметры сервера CTSA: протокол, IP-адрес и порт, сохранить настройки (например: HTTP, 10.20.1.123, 8080).

7. ЛИЦЕНЗИРОВАНИЕ СИСТЕМЫ

Для корректного запуска Системы необходима действующая лицензия, выпущенная на сервере лицензирования производителя ООО «Кросстех Солюшнс Групп» для данной организации.

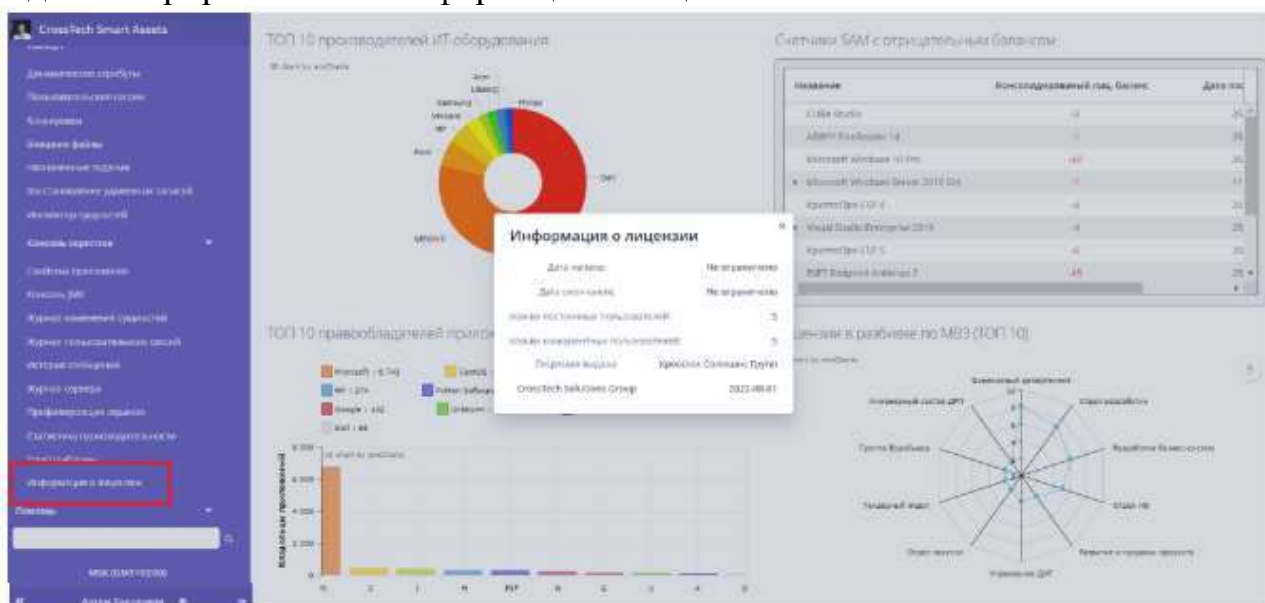
Лицензия на Систему имеет следующие параметры:

- Дата начала. Дата начала действия лицензии (не обязательный параметр). Если не указано, то срок не ограничен;
- Дата окончания. Дата окончания действия лицензии (не обязательный параметр). Если не указано, то срок не ограничен;
- Кол-во постоянных пользователей. Кол-во пользователей, которым доступен для выбора тип доступа Постоянный. Такие пользователи всегда могут произвести вход в Систему;
- Кол-во конкурентных пользователей. Кол-во пользователей, которые могут одновременно быть подключены к Системе, имея тип доступа Конкурентный. Такие пользователи могут произвести вход в Систему, только если текущее количество одновременных конкурентных подключений не превышает общего количества доступных. При этом тип доступа Конкурентный может иметь неограниченное количество пользователей (данный тип доступа устанавливается по умолчанию);
- Лицензия выдана. Юридическое лицо, кому выдана лицензия;
- Когда и кем выдана. Юридическое лицо, выпустившее лицензию (CrossTech Solutions Group) и дата выпуска данной лицензии.

На административного пользователя admin лицензией не накладываются ограничения на вход: под данным пользователем всегда можно подключиться к Системе.

Для просмотра информации о лицензии в Системе необходимо открыть пункт меню:

Администрирование → Информация о лицензии.

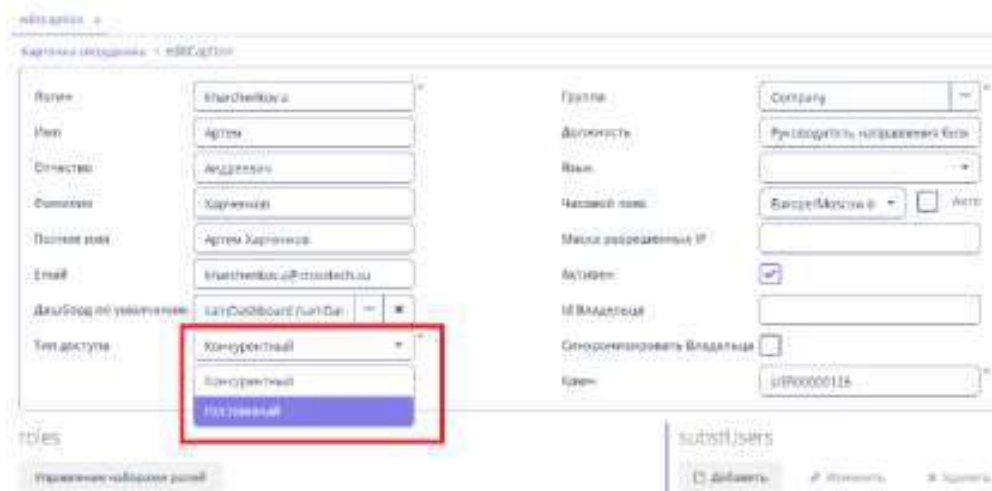


Выдача лицензий пользователям

Для выдачи лицензий пользователям необходимо перейти в экран Пользователей (именно Пользователей, а не Сотрудников, так как будет менять тип доступа учётной записи). Для этого нужно открыть пункт меню: Администрирование → Пользователи, найти соответствующего пользователя и перейти в его карточку.

Изменить Тип доступа на один из двух вариантов:

- **Постоянный.** Когда будет выдано максимальное количество постоянных доступов (указанное в лицензии на Систему), данный тип доступа станет недоступным для назначения;
- **Конкурентный.** Можно назначить всегда.



При достижении максимального количества одновременных подключений пользователю будет выведена ошибка входа с указанием причины. В данном случае необходимо связаться с администратором для изменения типа доступа, либо закупки дополнительных лицензий.

Установка лицензии на сервер приложений

При развёртывании приложения через WAR-архив необходимо разместить лицензию `license.ctsa` и публичный ключ `public.key` в директории `<tomcat_path>\work\app_home\am`, например:

Пример 1. Директория для размещения лицензии при WAR-развёртывании

```
C:\Program Files\Apache Software Foundation\Tomcat 9.0\work\app_home\am
```

При развёртывании приложения через UberJar-файл необходимо разместить лицензию `license.ctsa` и публичный ключ `public.key` в директории `<workdir_with_jar_file>\am`, например:

Пример 2. Директория для размещения лицензии при UberJar-развёртывании

```
C:\am_workdir\am
```

При локальном развёртывании приложения через Cuba Studio (для разработки) необходимо разместить лицензию `license.ctsa` и публичный ключ `public.key` в директории `<project_dir>\deploy\app_home\am`, например:

Пример 3. Директория для размещения лицензии при развёртывании из Cuba Studio

C:\Users\kharchenkov.a\StudioProjects\crosstech_am\deploy\app_home\am

Проверка лицензии

Система проверяет наличие и срок действия лицензии при старте приложения. В случае, если лицензия не найдена (или файл повреждён), в системный лог будет выведено следующее сообщение:

Caused by: com.crosstech.am.exception.LicenseException: Ошибка открытия файла ключа ...

В случае, если срок действия лицензии истёк (или ещё не начался), в системный лог будет выведено следующее сообщение:

Caused by: org.springframework.beans.BeanInstantiationException: Failed to instantiate [com.crosstech.am.core._common.LicenseProviderBean]: Constructor threw exception; nested exception is com.crosstech.am.exception.LicenseException: Срок лицензии истек или еще не начался. Обратитесь к поставщику лицензии ...

В обоих вышеперечисленных случаях приложение не будет запущено.

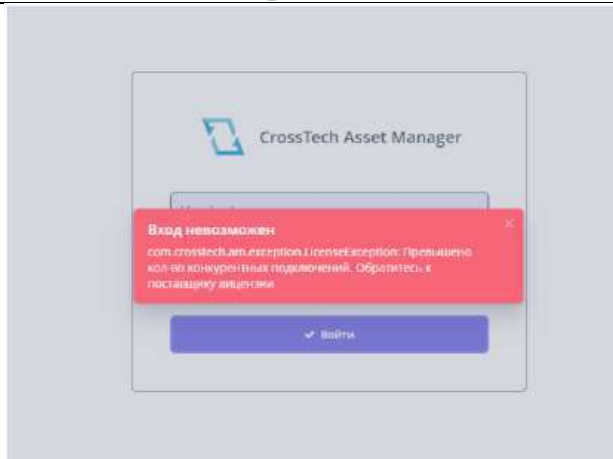
В случае, если лицензия активна и не повреждена (содержимое файла с момента подписи не изменено), в системный лог будет выведено следующее сообщение:

14:39:34.965 INFO c.s.a.c._common.LicenseProviderBean - Файл ключа успешно прочитан!
14:39:34.986 INFO c.s.a.c._common.LicenseProviderBean - Файл лицензии успешно прочитан!
14:39:34.992 INFO c.s.a.c._common.LicenseProviderBean - Верификация лицензии успешно пройдена!

В данном случае приложение будет запущено правильно.

При входе пользователя с конкурентным типом доступа будет проверено наличие свободных слотов для подключения. Если слотов больше нет, пользователю будет выведено следующее сообщение:

Caused by: com.crosstech.am.exception.LicenseException: Превышено кол-во конкурентных подключений. Обратитесь к поставщику лицензии



8. ТРЕБОВАНИЯ К АППАРАТНОМУ ОБЕСПЕЧЕНИЮ

Ниже представлены минимальные требования к аппаратному обеспечению с учетом количества одновременно работающих пользователей. Также в данные требования включены минимальные требования к операционной системе.

Таблица 1. Кол-во одновременных подключений: 5

| Назначение сервера | Процессор | Память | Диск | |
|------------------------|-----------|--------|------|---------|
| Сервер приложения и БД | 2 core | 3GB | SATA | 7200rpm |

Таблица 2. Кол-во одновременных подключений: 10

| Назначение сервера | Процессор | Память | Диск | |
|------------------------|-----------|--------|------|---------|
| Сервер приложения и БД | 2 core | 4GB | SATA | 7200rpm |

Таблица 3. Кол-во одновременных подключений: 25

| Назначение сервера | Процессор | Память | Диск | |
|------------------------|-----------|--------|------|---------|
| Сервер приложения и БД | 2 core | 6GB | SATA | 7200rpm |

Таблица 4. Кол-во одновременных подключений: 50

| Назначение сервера | Процессор | Память | Диск | |
|------------------------|-----------|--------|------|---------|
| Сервер приложения и БД | 4 core | 6GB | SATA | 7200rpm |

Таблица 5. Кол-во одновременных подключений: 75

| Назначение сервера | Процессор | Память | Диск | |
|------------------------|-----------|--------|------|----------|
| Сервер приложения и БД | 4 core | 8GB | SAS | 15000rpm |

Таблица 6. Кол-во одновременных подключений: 100

| Назначение сервера | Процессор | Память | Диск | |
|------------------------|-----------|--------|------|----------|
| Сервер приложения и БД | 4 core | 10GB | SAS | 15000rpm |

Таблица 7. Кол-во одновременных подключений: 125

| Назначение сервера | Процессор | Память | Диск | |
|------------------------|-----------|--------|------|----------|
| Сервер приложения и БД | 4 core | 12GB | SAS | 15000rpm |

Таблица 8. Кол-во одновременных подключений: 150

| Назначение сервера | Процессор | Память | Диск | | |
|--------------------|-----------|--------|------|----------|--------|
| Сервер приложений | 4 core | 6GB | SATA | 7200rpm | |
| Сервер БД | 2 core | 4GB | SAS | 15000rpm | RAID10 |

Таблица 9. Кол-во одновременных подключений: 200

| Назначение сервера | Процессор | Память | Диск | | |
|--------------------|-----------|--------|------|----------|--------|
| Сервер приложений | 4 core | 8GB | SATA | 7200rpm | |
| Сервер БД | 2 core | 6GB | SAS | 15000rpm | RAID10 |

Таблица 10. Кол-во одновременных подключений: 250

| Назначение сервера | Процессор | Память | Диск | | |
|--------------------|-----------|--------|------|----------|--------|
| Сервер приложений | 4 core | 10GB | SATA | 7200rpm | |
| Сервер БД | 2 core | 8GB | SAS | 15000rpm | RAID10 |

Таблица 11. Кол-во одновременных подключений: 300

| Назначение сервера | Процессор | Память | Диск | | |
|--------------------|-----------|--------|------|----------|--------|
| Сервер приложений | 4 core | 12GB | SATA | 7200rpm | |
| Сервер БД | 4 core | 10GB | SAS | 15000rpm | RAID10 |

Таблица 12. Кол-во одновременных подключений: 400

| Назначение сервера | Процессор | Память | Диск | | |
|--------------------|-----------|--------|------|----------|--------|
| Сервер приложений | 4 core | 14GB | SATA | 7200rpm | |
| Сервер БД | 4 core | 10GB | SAS | 15000rpm | RAID10 |

Таблица 13. Кол-во одновременных подключений: 500

| Назначение сервера | Процессор | Память | Диск | | |
|--------------------|-----------|--------|------|----------|--------|
| Сервер приложений | 4 core | 16GB | SATA | 7200rpm | |
| Сервер БД | 4 core | 12GB | SAS | 15000rpm | RAID10 |

Таблица 14. Кол-во одновременных подключений: 1000

| Назначение сервера | Процессор | Память | Диск | | |
|--------------------|-----------|--------|------|----------|--------|
| Сервер приложений | 8 core | 32GB | SATA | 7200rpm | |
| Сервер БД | 6 core | 16GB | SAS | 15000rpm | RAID10 |

Для развертывания Системы с количеством одновременных подключений больше 1000 необходимо использовать вариант с Кластером серверов приложений Web Client, работающих с одним сервером Middleware.

Таблица 15. Кол-во одновременных подключений: 1500

| Назначение сервера | Процессор | Память | Диск | | |
|----------------------------------|-----------|--------|------|----------|--------|
| Load Balancer | 2 core | 2GB | SATA | | |
| 2 * Сервер приложений Web client | 4 core | 20GB | SATA | 7200rpm | |
| Сервер приложения Middleware | 6 core | 10GB | SATA | 7200rpm | |
| Сервер БД | 8 core | 24GB | SAS | 15000rpm | RAID10 |

Таблица 16. Кол-во одновременных подключений: 2000

| Назначение сервера | Процессор | Память | Диск | | |
|----------------------------------|-----------|--------|------|----------|--------|
| Load Balancer | 2 core | 2GB | SATA | | |
| 2 * Сервер приложений Web client | 4 core | 24GB | SATA | 7200rpm | |
| Сервер приложения Middleware | 8 core | 12GB | SATA | 7200rpm | |
| Сервер БД | 12 core | 32GB | SAS | 15000rpm | RAID10 |

Таблица 17. Кол-во одновременных подключений: 3000

| Назначение сервера | Процессор | Память | Диск | | |
|----------------------------------|-----------|--------|------|----------|--------|
| Load Balancer | 2 core | 2GB | SATA | | |
| 2 * Сервер приложений Web client | 4 core | 24GB | SATA | 7200rpm | |
| Сервер приложения Middleware | 12 core | 18GB | SATA | 7200rpm | |
| Сервер БД | 16 core | 48GB | SAS | 15000rpm | RAID10 |

Для обеспечения полной отказоустойчивости приложения нужно использовать Кластер серверов приложений Web Client, работающих с кластером серверов Middleware.

В случае отказоустойчивого решения необходимо покупать лицензию на каждый Сервер приложения Middleware.

Таблица 18. Кол-во одновременных подключений: 100

| Назначение сервера | Процессор | Память | Диск | | |
|----------------------------------|-----------|--------|----------------------------|----------|--------|
| 2 * Load Balancer | 2 core | 2GB | SATA | | |
| 2 * Сервер приложений Web client | 2 core | 6GB | SATA | 7200rpm | |
| 2 * Сервер БД | 2 core | 4GB | SAS | 15000rpm | RAID10 |
| Файловое хранилище (SAN или NAS) | | | Зависит от объёма вложений | | |

Таблица 19. Кол-во одновременных подключений: 500

| Назначение сервера | Процессор | Память | Диск | | |
|----------------------------------|-----------|--------|------|----------|--------|
| 2 * Load Balancer | 2 core | 2GB | SATA | | |
| 2 * Сервер приложений Web client | 4 core | 15GB | SATA | 7200rpm | |
| 2 * Сервер приложений Middleware | 4 core | 6GB | SATA | 7200rpm | |
| 2 * Сервер БД | 2 core | 8GB | SAS | 15000rpm | RAID10 |

Таблица 20. Кол-во одновременных подключений: 1500

| Назначение сервера | Процессор | Память | Диск | | |
|----------------------------------|-----------|--------|----------------------------|----------|--------|
| 2 * Load Balancer | 2 core | 2GB | SATA | | |
| 4 * Сервер приложений Web client | 4 core | 20GB | SATA | 7200rpm | |
| 2 * Сервер приложений Middleware | 6 core | 10GB | SATA | 7200rpm | |
| 2 * Сервер БД | 8 core | 24GB | SAS | 15000rpm | RAID10 |
| Файловое хранилище (SAN или NAS) | | | Зависит от объёма вложений | | |

Таблица 21. Кол-во одновременных подключений: 3000

| Назначение сервера | Процессор | Память | Диск | | |
|----------------------------------|-----------|--------|----------------------------|----------|--------|
| 2 * Load Balancer | 2 core | 4GB | SATA | | |
| 6 * Сервер приложений Web client | 4 core | 24GB | SATA | 7200rpm | |
| 2 * Сервер приложений Middleware | 12 core | 24GB | SATA | 7200rpm | |
| 2 * Сервер БД | 16 core | 48GB | SSD | 15000rpm | RAID10 |
| Файловое хранилище (SAN или NAS) | | | Зависит от объёма вложений | | |

Ниже перечислены особенности, которые необходимо учитывать при развёртывании:

- Для лучшей производительности сервера СУБД желательно выделять оперативной памяти столько, чтобы туда смогла полностью поместиться БД.
- При настройке выделяемой оперативной памяти для СУБД и Системы необходимо учитывать, чтобы для ОС было зарезервировано минимум 2GB. При несоблюдении данного условия возможны проблемы с производительностью.
- При расчете объема жесткого диска под СУБД можно ориентироваться на усредненный размер документа и его связей равный 40kB, т.о. для хранения 100000 документов в СУБД необходимо минимум 4GB.
- Данные расчеты произведены на процессорах с тактовой частотой не меньше 2,5 ГГц.
- Размер жестких дисков под файловое хранилище зависит от предполагаемого использования системы и среднего размера файла, прикрепляемого к карточке.

Пример 4. Расчёт размера жесткого диска.

Усредненный размер документа и его связей равен 40kB, т.о. для хранения 100000 документов в СУБД необходимо минимум 4Гб. Размер жестких дисков под файловое хранилище зависит от предполагаемого использования системы и среднего размера файла, прикрепляемого к карточке. Если к одной карточке прикладываются вложения общим объемом в 5 мб, то для хранения в расчете на 100000 документов понадобится ~500Гб.

9. ЭКСПЛУАТАЦИЯ СИСТЕМЫ

9.1 Жизненный цикл Актива

Модуль, описывающий процедуры, относящиеся непосредственно к Активам и сопровождающие их на протяжении всего жизненного цикла: Актив (или ИТ-актив) – это любой материальный или нематериальный элемент ИТ-инфраструктуры, имеющий финансовое значение для организации. В Системе используется название Актив, так как в качестве учётной единицы могут учитываться как элементы ИТ-инфраструктуры, так и имущество административно-хозяйственной деятельности (АХД) и материально-технические ресурсы (МТР).

Активы могут быть материальными (МА) и нематериальными (НМА). К материальным активами (ИТ-оборудованию) относятся активы с типами:

- Персональные компьютеры;
- Мобильные компьютеры;
- Печатающие устройства;
- Периферийные устройства;
- Сетевое оборудование;
- Серверное оборудование;
- Расходные материалы и комплектующие;
- Системы хранения данных;
- и другие.

Материальные активы – это физические устройства, которые имеют реальные размеры и форму. Их можно перенести, потрогать и т.д.

К нематериальным активам относятся:

- Лицензии ПО;
- Сертификаты.

Нематериальные активы – не имеют физической формы, но представляют ценности для компании. Ими также можно управлять, но лишь "виртуально", например, назначая лицензию на пользователя.

В списке Активы должны выводиться все заведённые в систему Активы, вне зависимости от типа и статуса.

По умолчанию на экране ИТ-оборудование должен быть доступен фильтр по имени хоста, типу, статусу и др., а на экране лицензий должен быть доступен фильтр по наименованию и сотруднику. Карточка Актива должна меняться в

зависимости от его типа и статуса, например для Компьютера на складе будет виден один набор вкладок, в то время как для Лицензии в эксплуатации набор вкладок будет отличаться.

Активы могут находиться в разных статусах. Статусная модель Активов является одинаковой вне зависимости от Типа. В таблице ниже приведены статусы Активов с описанием и допустимыми переходами между ними:

| Статус | Описание | Возможен переход в статусы |
|----------------|--|---|
| В эксплуатации | Актив используется и закреплён за пользователем | На складе, В ремонте |
| На складе | Актив в данный момент не используется и находится на складе | В эксплуатации, В аренде, В ремонте, Списан |
| В аренде | Актив в данный момент находится в аренде у другого предприятия | На складе |
| В ремонте | Актив в данный момент неисправен и отправлен в ремонт | На складе, В эксплуатации |
| Списан | Актив в данный момент списан из учёта | На складе, Утилизирован |
| Утилизирован | Актив полностью выведен из эксплуатации и утилизирован | - |

События жизненного цикла

События жизненного цикла – это сущности, которые генерируются и привязываются к Активу, когда происходит какое-либо значимое событие в его жизненном цикле.

События жизненного цикла должны быть разделены на следующие виды:

1. Поставка

Данное событие возникает, когда происходит приёмка одного или нескольких Активов по Договору на склад.

2. Перемещение

Данное событие возникает, когда один или несколько Активов перемещаются с помощью мастера перемещения.

3. Инвентаризация

Данное событие возникает, когда по одному или нескольким Активам вводятся результаты инвентаризации.

Информационные системы

Система должна предоставлять возможность учитывать Информационные системы компании.

Информационная система — сущность, определяющая систему, предназначенную для хранения, поиска и обработки информации, и соответствующие организационные ресурсы (человеческие, технические, финансовые и т. д.), которые обеспечивают и распространяют информацию. В состав Информационной системы должны входить Активы разных типов:

- Лицензии;
- Серверы;
- Компьютеры;
- СХД;
- другие.

Основные средства/комплекты

Система должна предоставлять возможность учитывать и вести Основные средства.

Основные средства — материальные активы предприятия для различных нужд, закупочная стоимость которых выше 40000р., а ожидаемый срок полезного использования (эксплуатации) более одного года (или операционного цикла, если он длится дольше года). Стоимость основных средств за вычетом накопленной амортизации называется чистыми

основными средствами или остаточной стоимостью. Комплектом считается объединение Активов, которому ещё не присвоен инвентарный номер.

Активы следующих типов могут привязываться к карточке основного средства/комплекта:

- Лицензии;
- Серверы;
- Компьютеры;
- СХД;
- другие.

Входящие в состав Основного средства Активы называются Составом основного средства. У всего состава должен быть одинаковый с Основным средством инвентарный номер.

Основные средства/комплекты должны входить в сервисно-ресурсную модель (СРМ) и финансово-ресурсную модель (ФРМ) и должны быть связаны с Информационными системами связью типа клиент-ресурс.

Основное средство/комплект может находиться в разных статусах:

| Статус | Описание |
|-----------------|---|
| Комплектование | Процесс первичного объединения Активов в основное средство/комплект и создание карточки |
| В эксплуатации | Основное средство/комплект используется, состав сформирован |
| Частично списан | Основное средство/комплект ещё используется, но часть состава выведена из эксплуатации |

| | |
|--------|--|
| Списан | Основное средство/комплект выведено из эксплуатации в полном составе |
|--------|--|

9.2 Мастер приемки активов

Мастер приемки активов — мастер, позволяющий при помощи нескольких шагов осуществить приемку Активов по Договору, сформировать Документы приёмки и завести Активы в систему.

Система должна отображать следующие параметры приёмки:

- На склад — Склад, на котором будут размещены Активы после приемки;
- Ответственный/МОЛ — Сотрудник, ответственный за принимаемые Активы;
- На проект — целевой Проект, для которого приобретаются Активы;
- На статью расходов — Статья расходов, на которую будут списаны Расходы на покупку данного Актива;
- На МВЗ — Место возникновения затрат, на которое будут списаны Расходы на покупку данного Актива;
- Автор — Сотрудник, принимающий Активы;
- Дата/время события — дата/время приемки;
- Комментарий — дополнительная информация.

Помимо общих параметров должны быть параметры, которыми можно манипулировать в таблице с каждой принимаемой Позицией договора отдельно:

- К получению — кол-во принимаемых Активов в данной поставке (не может быть больше общего оставшегося кол-ва);
- Принять поэлементно — параметр, способный разделить партию принимаемых активов. Если параметр выбран, то все N принимаемых Активов данной Позиции должны быть созданы как N отдельных записей в базе данных с кол-вом 1шт, если данный признак не выбран, то должна быть создана одна запись в базе данных (партия) с количеством N;
- Цена за единицу с НДС, без НДС и др... — цена Позиции за единицу (должна быть возможность скорректировать);
- Сумма с НДС, без НДС и др... — общая стоимость Позиции (должна рассчитываться автоматически);
- Склад — Склад, на котором будут размещены Активы после приемки;

- Статья расходов — Статья расходов, на которую будут списаны Расходы на покупку данного Актива.

Данные по каждому активу должны автоматически передаваться в Документ приёмки активов со следующими параметрами:

- Вид — проставляется Приёмка автоматически;
- Номер — индивидуальный номер документа поставки;
- Валюта — Валюта, в которой производится расчет;
- Суммы — суммы с учетом и без учета НДС в валюте договора;
- Дата — дата формирования документа;
- Курс — курс валюты договора в российских рублях (курс указывается на дату формирования документа);
- Суммы — рублевый эквивалент сумм в валюте; данные параметры пересчитываются согласно правилам;
- Счет-фактура получен — параметр проставляется в случае, если при приемке был получен счет-фактура;
- Дата счета-фактуры — дата в счете-фактуре;
- Номер счета-фактуры — индивидуальный номер счета-фактуры;
- Скан-копия — Вложения (отсканированные документы приемки).

Мастер перемещения активов

Мастер перемещения активов — мастер, позволяющий при помощи нескольких шагов, произвести задокументированное перемещение актива (например, со склада в эксплуатацию), изменив его Статус и расположение.

После выполнения Перемещения активов для каждого Перемещённого актива должно быть сформировано событие жизненного цикла Перемещение.

При одновременном перемещении нескольких Активов, необходимо, чтобы они находились в одном Статусе и на одном Складе / у одного Сотрудника (в зависимости от Статуса).

Проведение инвентаризации

Инвентаризация – это проверка наличия имущества организации и состояния её активов на определённую дату путём сличения фактических данных с данными управленческого учёта.

Инвентарный номер – это индивидуальный номер, который присваивается объекту основных средств для организации учета и обеспечения контроля за сохранностью.

- Генерация инвентарных номеров

В Системе должна присутствовать возможность генерировать/обновлять инвентарные номера активов, которые недавно были приняты к учёту.

Таблица активов должна поддерживать множественный выбор, что позволит генерировать инвентарные номера сразу для большого количества активов.

Инвентарный номер должен быть сгенерирован только для активов с количеством 1, а также для активов, подлежащих инвентаризации.

- Печать инвентарных номеров

После того как инвентарные номера были сгенерированы, должна быть возможность распечатать их из Системы для дальнейшей расклейки на устройства.

Таблица активов должна поддерживать множественный выбор, что позволит печатать инвентарные номера сразу для большого количества активов.

- Приказы на инвентаризацию

Приказ на инвентаризацию – это формальное или неформальное распоряжение на проведение инвентаризации Активов какой-либо категории, местоположения и т. д. Приказы на инвентаризацию должны создаваться в системе как отдельные сущности, в рамках которых затем проводится инвентаризация.

В карточке Приказа на инвентаризацию должны быть доступны следующие вкладки:

- Шапка. Общая информация: ключ, дата создания, название, владелец и др;
- Исполнители. Информация о сотрудниках, которые будут выполнять инвентаризацию.

К одному Приказу могут быть привязаны несколько исполнителей, а каждый исполнитель может быть прикреплен к нескольким Приказам;

- Инвентаризационная ведомость.

Список Активов, подлежащих инвентаризации в рамках данного приказа. К одному Приказу в рамках инвентаризационной ведомости могут быть привязаны несколько Активов, а каждый Актив может быть прикреплен к нескольким Приказам.

В системе должен присутствовать фильтр для поиска результатов по типам:

- Ещё не проинвентаризированы. Результаты по Активам ещё не вводились;
- Отсутствующие активы. В рамках данной инвентаризации Актив не был найден;
- Найденные активы. В рамках данной инвентаризации Актив найден.
 - Формирование инвентаризационной ведомости

Перед проведением инвентаризации необходимо сформировать инвентаризационную ведомость, т.е. определить Активы, которые должны быть проинвентаризированы в рамках данного Приказа. Для этого на вкладке Инвентаризационная ведомость необходимо с помощью кнопки Добавить добавить в список необходимые Активы. При необходимости можно воспользоваться фильтром на экране Активов. В Инвентаризационную ведомость можно добавить только Активы, подлежащие инвентаризации.

После формирования ведомости должна присутствовать возможность выгрузить получившийся список в Excel.

- Ввод результатов инвентаризации

После прохода со считывателем штрих-кода по помещению и фиксации результатов на любом носителе (бумажном, в телефоне, в самом считывателе) необходимо внести информацию об инвентаризации в систему.

По завершении работы мастера в таблице результатов у соответствующих активов должны автоматически заполняться колонки Дата инвентаризации, Результат.

В карточке каждого из проинвентаризированных Активов на вкладке Жизненный цикл должно отображаться данное событие инвентаризации.

Если при вводе результата инвентаризации для Актива было указано, что Актив не найден, то данная информация должна отобразиться в списке результатов с соответствующей цветовой индикацией.

Для удобства поиска в списке результатов на вкладке Результаты инвентаризации должен быть предусмотрен фильтр, который по умолчанию включает в себя поисковую строку по инвентарному номеру.

Разделение партии активов

Активы должны учитываться в системе как поэлементно (когда позиция каждого Актива ведётся в таблице Активы как индивидуальный элемент учёта с количеством 1 шт), так и партией (когда в одной карточке в таблице Активы содержится более одного актива или партия, в этом случае количество в одной позиции > 1 шт).

В системе должна присутствовать возможность разделения Активов для ситуаций, когда в момент приемки они заносятся в систему единой записью с количеством больше 1, но в дальнейшем необходимо отделить и учитывать часть активов отдельно от текущей партии (например, для перемещения, для присвоения инвентарных или серийных номеров, выдачи лицензии и т.д.).

9.3 Управление соглашениями

В данном разделе описаны процедуры, относящиеся непосредственно к управлению соглашениями в Системе. Под соглашениями понимаются документы, являющиеся:

- Договорами;
- Счетами;
- Авансовыми отчётами;
- Дополнительными соглашениями;
- и др.

Договоры

По умолчанию на экране Договоров должен быть доступен фильтр по номеру. Под Договором в Системе понимается сущность, которая по своей сути может являться как самим договором, так и счётом, авансовым отчётом, доп. соглашением и др. Данная сущность может являться основанием для Приёмки по ней активов и имеет поведение (с точки зрения управления), схожее с сущностью Договор. В карточке Договора должна быть доступна следующая информация:

- Шапка. Общая информация: ключ, номер, владелец и др.
- Основная информация. Информация о сроках, иерархии, типах, стоимости, предмете и контрагенте.

- **Дополнительные соглашения.** Список доп. соглашений и субдоговоров, относящихся к данному Договору. С одним Договором может быть связано много доп. соглашений, а доп. соглашение может относиться лишь к одному Договору.
- **Позиции договора.** Список позиций договора (строк спецификации), относящихся к данному договору. С одним Договором может быть связано много позиций, а Позиция может относиться лишь к одному Договору.
- **Счета.** Список счетов, относящихся к данному договору. С одним договором может быть связано много Счетов, а Счёт может быть привязан лишь к одному Договору.
- **Документы.** Список документов, относящихся к данному договору. С одним Договором может быть связано много документов, а Документ может быть привязан лишь к одному Договору.
- **Активы.** Список активов, относящихся к данному договору. В данный список попадают активы, относящиеся ко всем документам, в свою очередь относящиеся к текущему договору.
- **События.** Список событий, относящихся к данному договору. С одним Договором может быть связано много событий, а Событие может относиться лишь к одному Договору.
- **Вложения.** Список вложений.
- **История изменений.** Список изменений атрибутов карточки.

Договоры могут быть разных типов и в разной валюте.

Позиции договора

Позиция договора – это сущность, которая определяет сколько и каких наименований должно быть принято по данному договору. Позиция договора также определяет цену за единицу, а также суммы с учетом ставок НДС. В позиции договора также можно задать некоторые параметры приёмки, например: признак поэлементного учёта или склад/статью расходов.

Дополнительные соглашения

Дополнительное соглашение – это договор с типом "Дополнительное соглашение". Они должны быть привязаны к основным договорам и расположены на втором уровне иерархии.

Счета

Счёт – это сущность, которая определяет оплату по тому или иному договору. Со счетом могут быть связаны документы поставки и заявки на оплату. Счета могут быть заведены в разных валютах.

Заявки на оплату

Заявка на оплату – это сущность, описывающая плановую или фактическую оплату по счёту.

Если дата оплаты в заявке указана, то она должна считаться оплаченной. Если валюта в Заявке на оплату отличается от Рубля, то курс оплаты рассчитывается исходя из текущего курса указанной валюты на дату оплаты.

Заявка может быть фактической и авансовой. Фактическая заявка определяет постоплату.

Авансовая заявка определяет оплату, которая производится до начала исполнения обязательств по договору. В зависимости от типов оплачиваемых заявок будут меняться правила расчёта сумм стоимостей и цен при приёмке активов.

В Заявке на оплату также можно указать Документ, который оплачивается данной заявкой.

Документы

Документы – это сущности, определяющие документацию, сопутствующую жизненному циклу Активов и Договоров. Документы могут быть разных видов: на поставку, на техподдержку, на продление лицензий, или на обновление ПО (на новую версию).

Документы могут являться накладными, актами, универсальными передаточными документами (УПД).

Документ должен содержать признак получения счета-фактуры. Документ должен быть связан со счётом или существовать отдельно от него. Если документ определяет поставку, он должен иметь ссылку на сущность Поставка. Суммы документов могут быть указаны в разных валютах. Документы могут иметь список связанных с ними Активов, например, для документа поставки это будет список поставленных активов, а для документа технической поддержки — это список активов, для которых приобреталась техподдержка.

События договора

События договора – Сущности, которые используются для контроля исполнения договора и отслеживания его мероприятий.

События должны иметь сроки "С" и "По" и описание. Если событие завершено, у него устанавливается дата окончания. Особо важные события должны помечаться признаком На контроле.

9.4 Управление финансами

В данном разделе описаны требования, относящиеся непосредственно к управлению финансами в Системе.

МВЗ

МВЗ — это сущность, описывающая место возникновения затрат. Сущность является иерархической. МВЗ группирует в себе Департаменты для создания 'центров учёта' Расходов, которые списываются на Активы этих Департаментов в процессе работы с ними. Таким образом, если Расход был списан на Актив какого-либо Департамента, то он должен автоматически учитываться в указанном в Департаменте МВЗ.

В карточке МВЗ должны быть доступны следующие вкладки и атрибуты:

- Шапка.
 - Ключ — автоматически генерируемое значение;
 - Название — наименование МВЗ;
 - Запись активна — признак активности записи;
- Основная информация.
 - Секция Иерархия — иерархическая информация по сущности;
- Департаменты. Здесь размещен список (с возможностью редактирования) Департаментов, относящихся к этому МВЗ.
- Расходы. Здесь выводится список Расходов, относящихся к этому МВЗ.

Статьи расходов

Статья расходов — сущность, позволяющая сгруппировать Расходы, которые списываются на Активы по каким-либо критериям. Сущность является иерархической.

Статьи расходов могут группировать Расходы следующих типов:

- Капитальные (CAPEX) — инвестиционные Расходы на покупку, модернизацию и ремонт физических активов, а также Расходы по обслуживанию кредитов на их приобретение;
- Операционные (ОРЕХ) — постоянные Расходы компании, связанные с ведением бизнеса (оплата лицензий, аренда, коммунальные платежи, услуги и др).

Статьи расходов:

В карточке Статьи расходов должны быть доступны следующие вкладки и атрибуты:

- Шапка.
 - Ключ — автоматически генерируемое значение;
 - Тип расходов — CAPEX или OPEX;
- Основная информация.
 - Название — наименование статьи расходов;
 - Описание/цель — полное описание статьи расходов;
 - Секция Иерархия — иерархическая информация по сущности;
- Расходы. Здесь отображается список Расходов, относящихся к этой Статье расходов.
- История изменений. Список изменений атрибутов карточки.

Расходы

Расходы — сущность, представляющая конкретные затраты по определенному Активу.

В карточке Расхода должны быть доступны следующие вкладки и атрибуты:

- Шапка.
 - Ключ — автоматически генерируемое значение;
 - Владелец — владелец Расхода;
 - Тип расходов — CAPEX или OPEX (подробнее: Статья расходов);

- Дата создания — дата появления Расхода;
- Сумма — сумма расхода в рублях;
- Основная информация.
- Название — наименование Расхода;
- МВЗ — ссылка на Место возникновения затрат;
- Статья расходов — ссылка на Статью расходов;
- Актив — ссылка на Актив, по которому был списан Расход;
- Секция Иерархия — иерархическая информация по сущности;
- История изменений. Список изменений атрибутов карточки.

Валюты и курсы

Валюта — сущность, описывающая денежную единицу какой-либо страны, а также тип денежной системы.

В Системе должны быть заведены основные валюты (список может быть расширен при необходимости):

- Доллар США;
- Российский рубль — валюта по умолчанию.

Сущность Валюта характеризуется следующими параметрами:

- Ключ — автоматически генерируемое значение.
- Краткое название — сокращенное название валюты;
- Полное имя — полное название валюты;
- Код валюты — код валюты;
- Числовой код валюты — международный числовой код валюты.

Многие сущности системы (Документы, Активы и др.), в которых присутствуют поля с ценой/стоимостью/суммами (или другие денежные поля), должны содержать также указание, в какой именно Валюте данная стоимость определяется.

В случае, если указанная Валюта отличается от российского рубля RUB, в большинстве случаев сущность содержит дополнительные поля, которые содержат в себе рублёвые эквиваленты указанных сумм, стоимостей и др. В случае, если Валюта = Российский рубль, то дополнительные поля с рублёвым эквивалентом также заполняются, но они равны значениям сумм в Валюте. Данные поля обычно расположены на одной вкладке для удобства сравнения и пересчёта.

Помимо вышеуказанных полей в расчёте стоимости в рублях участвует поле с курсом пересчёта. Данное поле не всегда берётся явно на дату пересчёта, иногда логика может быть более сложной (например, см. правила расчёта сумм, стоимостей и цен при приёмке активов). Курс пересчёта также отражается на вкладке со стоимостями и ценами и автоматически получается в рамках интеграции с сайтом ЦБ РФ.

Курс валюты на указанную дату в режиме 'онлайн' может быть получен только в случае постоянного подключения к интернету.

Ставки НДС

Ставка НДС — сущность, описывающая процент налога на добавленную стоимость. Используется для автоматического расчета сумм, учитывающих НДС в Системе.

Ставка НДС характеризуется следующими параметрами:

- Ключ — автоматически генерируемое значение;
- Системное имя — системный код;
- Название — название для отображения на экране;
- Процент — процент НДС;
- Коэффициент — числовое значение для умножения при формировании итоговой стоимости объектов системы с учётом НДС. Рассчитывается как $1 + \text{процент}/100$, например для ставки 20% коэффициент будет равен 1,2.

В Системе должны быть заведены основные Ставки НДС, действующие в Российской Федерации (при необходимости список можно расширить):

- НДС: 0%;
- НДС: 10%;
- НДС: 20%;

- Без НДС.

9.5 Управление делами

В данном разделе описаны процедуры и сущности, относящиеся непосредственно к управлению делами компании.

Дела – это сущности, которые определяют ITSM-объекты, необходимые к обработке и выполнению. Под Делами в системе понимается базовая сущность, от которой наследуются сущности 3-х видов:

- Заявка;
- Запрос на обслуживание (ЗНО, Запрос);
- Наряд на работу (Наряд).

В Системе учитываются и ведутся только Дела, относящиеся к процессу управления Активами. К Делах должна быть привязана функциональность BPM-процессов. При создании любого Дела автоматически создаётся новый экземпляр BPM-процесса, в рамках которого идёт жизненный цикл данного Дела, изменяя статусы. Для каждого из видов Дел (Заявка, Запрос, Наряд) BPM-процесс отдельный и будет рассмотрен в соответствующем разделе.

У Дел всех видов присутствует большой набор одинаковых атрибутов, поэтому данный набор атрибутов вынесен в базовую сущность Дела:

- Ключ — автоматически генерируемое значение;
- Владелец — владелец Дела;
- Дата/время создания — дата и время, когда Дело было создано Автором;
- Автор — Сотрудник, создавший Дело;
- Описание — предмет Дела;
- Статус — Статус Дела;
- Тип — тип Дела (Заявка, Запрос, Наряд);
- Комментарий — комментарии и справочная информация;
- Приоритет — приоритет выполнения;
- Назначенный — ответственный за выполнение Дела Сотрудник;

- Рабочая группа — ответственная за выполнение Дела Рабочая группа;
- Вложения — список Вложений;
- BPM-процесс — BPM-процесс, определяющий жизненный цикл Дела;
- Базовая заявка — Заявка, от которой было создано данное Дело.

Статусы дел

Дела могут находиться в разных статусах. Статусная модель Дел является одинаковой вне зависимости от вида, т.е. является ли данное дело Заявкой, Запросом или Нарядом.

Переходы между статусами осуществляется и определяются с помощью BPM-процесса. В таблице ниже приведены статусы Дел с описанием этих статусов:

| Статус | Описание |
|----------------|--|
| Новое | Дело только что было создано, пока ещё не прошло классификацию и категоризацию |
| Согласование | Дело в данный момент находится на этапе согласования с ответственным лицом или группой лиц |
| В работе | Ответственное лицо или группа лиц в данный момент работает над выполнением данного Дела |
| Приостановлено | В данный момент работа над Делом приостановлена по каким-либо причинам. Дело ожидает устранения этих причин и возобновления работы |
| Отклонено | Ответственное лицо или группа лиц не согласовала выполнение данного Дела |

| | |
|-----------|---|
| Выполнено | Ответственное лицо или группа лиц выполнила все работы по данному Делу. Дело ожидает подтверждения выполнения работ автором/ответственным за Дело |
| Закрото | Выполненные работы по Делу были подтверждены автором/ответственным за Дело и данное Дело перемещено в архив как закрытое |

Заявки

Заявка — это Дело, определяющее обращение Сотрудника по какому-либо вопросу по любому из каналов (телефон, портал самообслуживания, почта и др.). Заявка подлежит категоризации Рабочей группой первой линии поддержки и может быть классифицирована как:

- Запрос на обслуживание;
- Инцидент;
- Изменение;
- Жалоба;
- Благодарность;
- и др.

Для Заявок реализованы несколько представлений, при открытии которых можно быстро отфильтровать только необходимые Заявки:

- Заявки — все Заявки;
- Заявки, назначенные на меня — Заявки, у которых в поле Назначенный указан текущий Сотрудник;
- Заявки, назначенные на мою РГ — Заявки, у которых в поле Рабочая группа указана Рабочая группа, в которую входит текущий Сотрудник;

- Заявки, ожидающие моих действий — Заявки, у которых в текущем ВРМ-процессе активной является задача, ожидающая действий от текущего Сотрудника.

В карточке Заявки должны быть доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Владелец — владелец записи;
 - Вид — вид Заявки (Запрос на обслуживание, Инцидент, Изменение, Жалоба и др.)
 - Статус — Статус Заявки;
 - Тип — Тип Заявки;
- SLA. Вкладка, содержащая в себе информацию о дате/времени перехода в разные Статусы.
- Общая информация. Основные атрибуты:
 - Тема — краткое описание Заявки;
 - Описание — подробное описание Заявки;
 - Автор — Сотрудник, создавший Заявку;
 - Получатель — Сотрудник, для которого был создан Заявка;
 - Совпадает с автором — признак, отвечающий за то что автор и получатель совпадают;
 - Назначенный — ответственный за выполнение Заявки Сотрудник;
 - Рабочая группа — ответственная за выполнение Заявки Рабочая группа;
- Связанные объекты. Сущности (в большинстве случаев, Запросы), привязанные к данной Заявке в рамках Эскалации. С одной Заявкой может быть связано несколько Запросов, а Запрос может быть привязан только к одной Заявке;
- Вложения. Список вложений;

- Бизнес-процесс. Вкладка, на которой отображается диаграмма текущего ВРМ-процесса и список активных/завершённых задач;
- История изменений. Список изменений атрибутов карточки.

Для Заявки создана одна универсальная модель ВРМ-процесса. При создании Заявки, стартует экземпляр процесса, указанный в соответствующем Типе Заявки.

Запросы на обслуживание

Запрос на обслуживание (ЗНО, Запрос) — это Дело, определяющее стандартную (или нестандартную) последовательность действий, которые необходимо выполнить Рабочей группе или конкретному Сотруднику для удовлетворения какой-либо потребности автора.

Запрос систематизирует действия, выполняемые в Системе и вне её в рамках конкретной процедуры.

Для Запросов реализованы несколько представлений, при открытии которых можно быстро отфильтровать только необходимые Запросы:

- Запросы на обслуживание — все Запросы;
- ЗНО, назначенные на меня — Запросы, у которых в поле Назначенный указан текущий Сотрудник;
- ЗНО, назначенные на мою РГ — Запросы, у которых в поле Рабочая группа указана Рабочая группа, в которую входит текущий Сотрудник;
- ЗНО, ожидающие моих действий — Запросы, у которых в текущем ВРМ-процессе активной является задача, ожидающая действий от текущего Сотрудника.

В карточке Запроса доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Владелец — владелец записи;
 - Дата создания — дата и время, когда Запрос был создан Автором;
 - Вид — является ли Запрос стандартным или нестандартным;
 - Статус — Статус Запроса;

- Приоритет — приоритет выполнения;
- Общая информация. Основные атрибуты:
 - Тема — краткое описание Запроса;
 - Описание — подробное описание Запроса;
 - Тип — Тип Запроса;
 - Заявка — Заявка, от которой был создан данный Запрос;
 - Автор — Сотрудник, создавший Запрос;
 - Получатель — Сотрудник, для которого был создан Запрос;
 - Назначенный — ответственный за выполнение Запроса Сотрудник;
 - Рабочая группа — ответственная за выполнение Запроса Рабочая группа;
 - Параметры запроса — блок Динамических атрибутов, настроенных для данного Типа Запроса;
- Наряды на работу. Наряды, привязанные к данному Запросу.

С одним Запросом может быть связано несколько Нарядов, а Наряд может быть привязан только к одному Запросу;

- Вложения. Список вложений;
- Бизнес-процесс. Вкладка, на которой отображается диаграмма текущего ВРМ-процесса и список активных/завершённых задач;
- История изменений. Список изменений атрибутов карточки.

Для Запроса на каждый его Тип создана отдельная модель ВРМ-процесса. В зависимости от того, какого типа Запрос создаётся, стартует экземпляр процесса, указанный в соответствующем Типе.

Список реализованных моделей ВРМ-процесса, соответствующие разным Типа Запроса, реализованных из коробки приведён ниже:

- ЗНО — Возврат активов из аренды;
- ЗНО — Возврат активов из ремонта;
- ЗНО — Передача активов в аренду;

- ЗНО — Передача активов в ремонт;
- ЗНО — Перемещение активов;
- ЗНО — Предоставление активов;
- ЗНО — Предоставление полномочий на ПО;
- ЗНО — Приёмка активов и постановка на учёт;
- ЗНО — Проведение инвентаризации;
- ЗНО — Списание активов;
- ЗНО — Утилизация активов.

Наряды на работу

Наряд на работу (Наряд) — это Дело, определяющее работу (или задание), которую необходимо выполнить ответственному Сотруднику.

Для Нарядов на работу реализованы несколько представлений, при открытии которых можно быстро отфильтровать только необходимые Наряды:

- Наряды на работу — все Заявки;
- Наряды, назначенные на меня — Наряды, у которых в поле Назначенный указан текущий Сотрудник;
- Наряды, назначенные на мою РГ — Наряды, у которых в поле Рабочая группа указана Рабочая группа, в которую входит текущий Сотрудник;
- Наряды, ожидающие моих действий — Наряды, у которых в текущем ВРМ-процессе активной является задача, ожидающая действий от текущего Сотрудника.

В карточке Наряда доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Владелец — владелец записи;
 - Статус — Статус Заявки;
 - Тип — Тип Заявки;

- **Приоритет** — приоритет выполнения;
- **Порядковый номер** — номер Наряда в рамках одного Запроса;
- **Дата создания** — дата и время, когда Запрос был создан Автором;
- **Основная информация. Основные атрибуты:**
 - **Запрос на обслуживание** — ссылка на связанный Запрос, в рамках которого выполняется данный Наряд;
 - **Описание запроса** — подробное описание связанного Запроса, в рамках которого выполняется данный Наряд;
 - **Описание** — подробное описание Наряда, по умолчанию копируется из Типа Наряда;
 - **Тип** — Тип Наряда;
 - **Автор** — Сотрудник, создавший Наряд;
 - **Назначенный** — ответственный за выполнение Наряда Сотрудник;
 - **Рабочая группа** — ответственная за выполнение Наряда Рабочая группа;
 - **Параметры** — блок Динамических атрибутов, настроенных для данного Типа Наряда;
- **Вложения. Список вложений;**
- **Бизнес-процесс.** Вкладка, на которой отображается диаграмма текущего ВРМ-процесса и список активных/завершённых задач;
- **История изменений.** Список изменений атрибутов карточки.

Для Наряда создана одна универсальная модель ВРМ-процесса. При создании Наряда, стартует экземпляр процесса, указанный в соответствующем Типе Наряд на работу.

Эскалация дел

Эскалация дел — процесс, в рамках которого описана последовательность создания Дел в Системе и автоматического создания новых Дел на основе существующих.

Эскалация заявок

Изначально, при возникновении потребности, Сотрудник заводит в систему Заявку. Затем, в соответствии с ВРМ-процессом Заявка проходит классификацию и либо берётся в работу любым Сотрудником назначенной Рабочей группы, либо отклоняется.

Если Заявка взята в работу, то следующим шагом ответственным за выполнение Сотрудником происходит оценка: необходимо ли создание связанного с данной Заявкой Запроса. В случае, если данную Заявку можно решить с помощью применения Запроса, то ответственный может выполнить эскалацию Заявки в Запрос:

- Заявка должна находиться в Статусе В работе, текущая задача Эскалация;
- Необходимо нажать кнопку Эскалировать в... на боковой панели экрана;
- В открывшемся мастере эскалации необходимо заполнить параметры сущности, в которую будет произведена эскалация и нажать Готово;
- После проведения эскалации на вкладке Связанные объекты появится связанный Запрос.

Эскалация запросов

В процессе выполнения некоторых Запросов ВРМ-процесс может автоматически создавать Наряды на работу и связывать их с исходным Запросом.

В данном случае создаются два параллельных Наряда и привязываются к Запросу.

Как и в случае с Эскалацией заявок с текущего момента необходимо, чтобы все связанные Наряды были переведены в статус Закрыто для того, чтобы текущий Запрос автоматически стал Выполненным.

9.6 Организационная структура

В данном разделе описаны процедуры и сущности, относящиеся непосредственно к организационно-штатной структуре компании.

Подразделения

Подразделение – это сущность, описывающая организационную единицу компании (департамент, отдел, направление и др.). Справочник Подразделений является иерархическим.

Подразделения должны загружаться в систему из Битрикс24.

В карточке Подразделения должны быть доступны следующие вкладки:

- Шапка. Основная информация, содержащая ключ, наименование и др;
- Основная информация. Информация об МВЗ, к которому относится Подразделение, код, информация об иерархии и техническая информация об ID в Битрикс24;
- Состав. Информация о Сотрудниках, входящих в состав Подразделения. Одно Подразделение может содержать несколько Сотрудников, а один Сотрудник может работать лишь в одном Подразделении;
- История изменений. Список изменений атрибутов карточки.

Сотрудники и пользователи

Сотрудник – это сущность, описывающая работника компании. Сотрудник может как иметь доступ в Систему, так и не иметь его. По своей сути Сотрудник является расширением сущности Пользователь и имеет все те же атрибуты, что и Пользователь, а также свои собственные дополнительные атрибуты.

По сути одна и та же сущность делится на две: Сотрудник — для управления бизнес-атрибутами, Пользователь — для управления учётной записью.

Сотрудники должны загружаться в систему из Active Directory и Битрикс24.

В карточке Сотрудника должны быть доступны следующие вкладки:

- Шапка. Основная информация, содержащая логин, ФИО, ключ, владельца и др;
- Общая информация. Информация о Подразделении, контакты, информация о размещении и техническая информация об ID в Битрикс24 (если есть);
- Владельцы. Информация о доступных для Сотрудника Владельцах. У одного сотрудника может быть несколько допустимых Владельцев;
- Портфель активов. Информация об Активах, закреплённых за данным Сотрудником (находящимся у него в использовании). Один Сотрудник может использовать несколько Активов разных Типов;

- Рабочие группы. Информация о Рабочих группах, в которых состоит данный Сотрудник. Один Сотрудник может состоять в нескольких Рабочих группах;
- История изменений. Список изменений атрибутов карточки.

Рабочие группы

Рабочая группа – это сущность, описывающая группу Сотрудников, объединённых между собой в соответствии с выполняемыми в Системе задачами.

В карточке Рабочей группы доступны следующие вкладки:

- Шапка. Основная информация, содержащая название, системное имя, ключ, и др;
- Состав рабочей группы. Информация о Сотрудниках, входящих в состав Рабочей группы. Одна Рабочая группа может содержать несколько Сотрудников, а один сотрудник может входить в несколько Рабочих групп.

9.7 Управление активами ПО (SAM)

Управление активами ПО (SAM) — это набор проверенных ИТ-практик, объединяющих людей, процессы и технологии для управления и оптимизации использования ПО в компании.

Управление активами ПО позволяет проводить аналитику и находить возможности для снижения расходов на приобретение ПО (достижение экономического эффекта) путём сопоставления количества закупаемых лицензий на ПО с количеством реально используемого, а также снизить риски несоответствия требованиям внешних и внутренних регуляторов (законодательство, отраслевые стандарты, внутренние политики), от которых зависит непрерывность бизнеса.

Система позволяет управлять активами ПО путём использования специальных сущностей и связанного с ними функционала.

Приложения

Приложение — сущность, определяющая конкретную установку программного обеспечения на устройство. Приложением считается экземпляр Наименования приложения, установленный на ИТ-оборудование в конкретную директорию в определённую дату.

Список Приложений чаще всего используется именно в привязке к ИТ-оборудованию, на котором оно было установлено.

В карточке Приложения доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Владелец — владелец записи;
 - Число пунктов — сколько позиций в лицензионном учёте занимает инсталляция данного Приложения;
- Техническая информация.
 - Компьютер — ИТ-оборудование, на котором установлено данное Приложение;
 - Наименование дискаверинга — наименование дискаверинга, определенное системой дискаверинга;
 - Эталонное наименование — наименование, к которому было нормализовано наименование дискаверинга;
 - Папка установки — директория, в которую непосредственно было установлено данное Приложение на ИТ-оборудование;
 - Дата установки — дата, когда Приложение было установлено на ИТ-оборудование;
 - Дата последней синхронизации с OCS — дата, когда Приложение было последний раз синхронизировано с OCS Inventory;
 - OCS ID — идентификатор записи в OCS Inventory.

Список Приложений наполняется автоматически, когда происходит интеграция с OCS Inventory. Новые Приложения (в привязке к ИТ-оборудованию) при синхронизации данных добавляются в таблицу, а уже существующие обновляются.

Лицензии

Лицензия — Актив, имеющий тип Лицензия ПО, относящийся к нематериальным. Лицензии являются Активами и хранятся в одной таблице с другими Активами разных Типов.

Лицензии содержат в себе информацию, подтверждающую лицензионные права на установку или использование какого-либо программного обеспечения. Лицензионные права могут быть разных типов. Какой именно тип прав указан у Лицензии, количество таких прав, сроки действия и другая важная информация указывается на вкладке Информация о лицензии в разделе Общие.

Информация о ключе активации, ID, аппаратном ключе и др. указывается на вкладке Информация о лицензии в разделе Активация и защита.

Лицензии используются в процедуре Управление активами ПО (SAM) для подсчёта суммарного количества прав на использование ПО в том или ином разрезе с помощью Счётчиков SAM.

Так как Лицензия является Активом, набор атрибутов и вкладок Лицензии определяется Типом Лицензия ПО и описан в разделе Активы. Лицензии также имеют Наименования, которые определяют что именно за Лицензия заведена в данной позиции.

Типы лицензий

Тип лицензии — сущность, определяющая, какого именно рода лицензионные права на программное обеспечение получает компания, приобретая Лицензию. Является иерархическим справочником.

Данный справочник создаётся автоматически при развёртывании Системы и содержит в себе основные Типы лицензии в соответствии с политиками лицензирования крупных вендоров. При необходимости данный справочник можно расширить, добавив собственные Типы лицензии.

Справочник Типов лицензий применяется в Управлении активами ПО (SAM) как атрибут Лицензии или Наименования лицензии.

В карточке Типа лицензий доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Название — сокращенное наименование местоположения;
 - Системное имя — системное наименование Типа лицензии;
- Основная информация. Информация об иерархии.

Правообладатели дискаверинга

Правообладатель дискаверинга — сущность, определяющая компанию или юрлицо, которое было обнаружено системой дискаверинга как имеющая юридические права на продукт или наименование. Данный справочник не является эталонным и может содержать в себе несколько названий на каждого учитываемого Правообладателя. Потребность в данном справочнике определяется тем, что системы дискаверинга могут по-разному определять одного и того же правообладателя (в зависимости от того, как он был записан в соответствующем поле).

Справочник Правообладателей дискаверинга содержит в себе все возможные наименования той или иной компании в привязке к эталонным Правообладателям.

Привязка осуществляется в процессе Нормализации данных и необходима для того, чтобы статистику по, например, количеству ПО определённого Правообладателя можно было получить корректно и агрегировано, без дублирования данных.

Справочник наполняется автоматически, когда происходит интеграция с OCS Inventory.

Новые Правообладатели, обнаруженные системой дискаверинга при синхронизации данных добавляются в таблицу, а уже существующие обновляются.

В карточке Правообладателя дискаверинга доступны следующие вкладки и атрибуты:

- Шапка.
- Ключ — автоматически генерируемое значение;
- Эталонный правообладатель — Правообладатель, к которому нормализован данный Правообладатель дискаверинга;
- Запись активна — признак активности записи;
- Название — название Правообладателя дискаверинга;
- Наименования. Список Наименований, относящихся к данному Правообладателю

дискаверинга. С одним Правообладателем дискаверинга может быть связано много Наименований, а Наименование может относиться лишь к одному Правообладателю дискаверинга.

Наименования дискаверинга

Наименование дискаверинга — сущность, определяющая наименование программного обеспечения, которое было обнаружено системой дискаверинга. Данный справочник не является эталонным и может содержать в себе несколько названий на каждого учитываемого Наименования.

Потребность в данном справочнике определяется тем, что системы дискаверинга могут по-разному определять одно и то же наименование (в зависимости от того, как оно было записано в соответствующем поле).

Справочник Наименований дискаверинга содержит в себе все возможные наименования того или иного программного обеспечения в привязке к эталонным Наименованиям приложений.

Привязка осуществляется в процессе Нормализации данных и необходима для того, чтобы статистику по, например, количеству ПО с определённым Наименованием можно было получить корректно и агрегировано, без дублирования данных.

Справочник наполняется автоматически, когда происходит интеграция с OCS Inventory.

Новые Наименования, обнаруженные системой дискаверинга при синхронизации данных добавляются в таблицу, а уже существующие обновляются.

В карточке Наименования дискаверинга доступны следующие вкладки и атрибуты:

- Шапка.
- Ключ — автоматически генерируемое значение;
- Эталонное наименование — Наименование приложений, к которому нормализовано данное Наименование дискаверинга;
- Код — код записи в системе дискаверинга;
- Название — название Наименования дискаверинга;

- Основная информация. Технические данные по программному обеспечению, имеющему данное наименование.
- Приложения. Список Приложений, относящихся к данному Наименованию дискаверинга. С одним Наименованием дискаверинга может быть связано много Приложений, а Приложение может относиться лишь к одному Наименованию дискаверинга.
- История изменений. Список изменений атрибутов карточки.

Нормализация данных

Нормализация данных — процесс организации данных в Системе, позволяющий 'привязать' разрозненные (или сырые) данные к эталонному справочнику. Данный процесс необходим для того, чтобы несколько записей из таблицы БД, обозначающих одно и то же, были привязаны к одной нормализованной записи для дальнейшего использования этой информации, например, в отчётах, во избежание дублирования информации и для возможности её консолидации. Нормализация активно применяется при синхронизации данных с OCS Inventory: обнаруженные системой дискаверинга 'сырые' наименования и правообладатели один раз приводятся к эталонным значениям, и, затем, при каждой последующей синхронизации ранее нормализованные 'сырые' данные будут автоматически привязываться к эталонным. Таким образом, нормализацию 'сырых' данных необходимо провести всего 1 раз.

Нормализация правообладателей

Нормализация правообладателей — процесс нормализации Правообладателей дискаверинга к эталонным Правообладателям.

Нормализация наименований

Нормализация наименований — процесс нормализации Наименований дискаверинга к эталонным Наименованиям приложений.

В отличие от правообладателей, при первичной загрузке из OCS Inventory Наименований дискаверинга на каждое из таких наименований дискаверинга автоматически создаётся эталонное Наименование, т.е. первичная нормализация данных происходит автоматически. Проводить нормализацию Наименований дискаверинга требуется только в случае, если нужно скорректировать их привязку к эталонному Наименованию.

Полномочия

Полномочия — сущность, определяющая, у каких Сотрудников или Подразделений присутствуют права на установку какого-либо программного обеспечения на свой компьютер.

Полномочия применяются в Управлении активами ПО (SAM) для решения задачи разграничения прав на установку ПО.

В карточке Полномочия доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Дата — дата выдачи данного полномочия на установку ПО;
 - Запрос на обслуживание — ссылка на Запрос на обслуживание, в рамках которого было выдано данное полномочие;
- Сотрудники. Информация о Сотрудниках, которым выданы права на установку в рамках данного полномочия. Одно полномочие может содержать несколько Сотрудников, а одному Сотруднику могут быть выданы права в рамках нескольких Полномочий;
- Подразделения. Информация о Подразделениях, которым выданы права на установку в рамках данного полномочия. Если выданы права на Подразделение, это значит, что права автоматически выдаются для всех его Сотрудников. Одно полномочие может содержать несколько Подразделений, а одному Подразделению могут быть выданы права в рамках нескольких Полномочий;
- Наименования. Информация о Наименованиях приложений, на которые выданы права на установку в рамках данного полномочия. Одно полномочие может содержать несколько Наименований приложений, а на одно Наименование могут быть выданы права в рамках нескольких Полномочий.

Управление полномочиями

Управление Полномочиями происходит на вкладке Полномочия сущности Наименования приложений. У каждого Наименования на данной вкладке возможно проставить признаки:

- Разрешено для всех. Признак указывает, что программное обеспечение с данным Наименованием могут устанавливать на свой компьютер все

Сотрудники. По умолчанию у всех Наименований приложений в Системе установлен данный признак.

Если данный признак установлен, то списки с допущенными в рамках полномочий Сотрудниками скрывается;

- Запрещено для всех. Признак указывает, что программное обеспечение с данным Наименованием не может устанавливаться на свой компьютер никто, кроме Сотрудников и Подразделений, которые добавлены в связанные Полномочия. По умолчанию у всех Наименований приложений в Системе данный признак НЕ установлен.

Если выбран один из данных признаков, то второй признак скрывается для выбора, т.к. они являются взаимоисключающими.

В комбинации с Полномочиями вышеуказанные признаки определяют критерий правомерности Наименования приложения по отношению к каждому установившему его к себе на компьютер Сотруднику. Критерии может принимать следующие значения:

- Запрещенная установка — критерий правомерности, когда Сотрудник не включён в Полномочия как допущенный и Наименование помечено как Запрещено всем;
- Разрешенная установка — критерий правомерности, когда Сотрудник либо включён в Полномочия как допущенный, либо Наименование помечено как Разрешено всем;
- Неавторизованная установка — критерий правомерности, когда Наименование не помечено как Разрешено всем/Запрещено всем, Сотрудник не включён в Полномочия как допущенный, но есть ещё хотя бы один сотрудник, который до установки данного Наименования допущен;
- Не определено — критерий правомерности, когда Наименование не помечено как Разрешено всем/Запрещено всем, Сотрудник не включён в Полномочия как допущенный и нет ни одного допущенного к установке данного Наименования ПО Сотрудника.

В Системе присутствует Отчёт "Правомерность использования ПО", который позволяет получить информацию по критериям правомерности установок Наименований приложений на компьютер для разных Сотрудников на основе настроек, выполненных на основании данного раздела.

Метрики SAM

Метрика SAM — сущность, определяющая логику подсчёта количества Приложений или Лицензий в Счётчике SAM. Содержит в себе запросы и скрипты, которые выполняются при нажатии на кнопку Подсчитать в Счётчике SAM.

Данный справочник создаётся автоматически при развёртывании Системы и содержит в себе основные метрики в соответствии с лучшими практиками управления активами ПО.

При необходимости данный справочник можно расширить, добавив собственные Метрики SAM.

В карточке Метрики SAM должны быть доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Запись активна — признак активности записи;
 - Название — название Метрики SAM;
 - Системное имя — системное название Метрики SAM;
- Параметры метрики.
 - Тип — атрибут, отвечающий за то, логику подсчёта чего определяет данная метрика
 - Объект — Объекты, в контексте которых будет производиться поиск Приложений или Лицензий: выполняться запрос, делаться подсчёт и т.д.;
- Скрипты.

Счётчики SAM

Счётчик SAM — сущность, являющаяся центральным объектом для сбора и вычисления статистики по количеству купленных лицензионных прав и фактически установленному ПО. В данной сущности представлено лицензионное соответствие по конкретному программному обеспечению и отображаются результаты подсчётов.

Данный справочник создаётся автоматически при развёртывании Системы и содержит в себе основные Счётчики SAM в соответствии с лучшими практиками управления активами ПО. При необходимости данный справочник можно расширить, добавив собственные Счётчики SAM. Счётчики SAM являются иерархической сущностью, иерархию определяет связь счётчиков предыдущей и следующей версии одного и того же программного обеспечения между собой.

Подсчёт значений в Счётчике SAM происходит на периодической основе, также в Системе присутствует возможность подсчёта значений вручную по нажатию на кнопку Рассчитать.

Для удобства и быстрого поиска нужных записей на экране по умолчанию предусмотрен фильтр по названию, а также фильтр Непустые, который при применении показывает только Счётчики SAM с ненулевыми результатами подсчёта по числу Приложений или Лицензий.

В карточке Счётчика SAM должны быть доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Запись активна — признак активности записи;
 - Глобальный — признак глобальности: если признак установлен, то Наименования лицензий и Наименования приложений, добавленные к подсчёту в данный Счётчик SAM не могут быть добавлены в другие Счётчики SAM;
 - Название — название Счётчика SAM;
 - Внешний Id — Id Счётчика SAM во внешней системе;
 - Дата последнего подсчёта — дата последнего подсчёта Счётчика SAM;
- Сводная информация. Результаты подсчётов.
 - Число прав и Число приложений — суммарные количества лицензионных прав и приложений соответственно, вычисленных счётчиком по заданным на вкладках Подсчёт лицензий и Подсчёт приложений правилам;
 - Лицензионный баланс — разница между количеством вычисленных лицензионных прав и приложений;

- Консолидированный лиц. баланс — количество вычисленных лицензионных прав с учётом передачи прав Счётчикам SAM предыдущих версий, а также с учётом переданных счётчиками обновления прав.
- Подсчёт лицензий. Вкладка, определяющая логику подсчёта количества лицензионных прав.
- Метрика SAM — Метрика SAM для подсчёта количества Лицензий;
- Наименования лицензий — Наименования лицензий, количество Лицензий по которым будут подсчитаны в рамках количества лицензионных прав. С одним Счётчиком SAM может быть связано много Наименований, а Наименование может подсчитываться в нескольких Счётчиках SAM.
- Подсчёт приложений. Вкладка, определяющая логику подсчёта количества установленных приложений.
- Метрика SAM — Метрика SAM для подсчёта количества Приложений;
- Наименования приложений — Наименования приложений, количество Приложений, по которым будут подсчитаны в рамках количества приложений. С одним Счётчиком SAM может быть связано много Наименований, а Наименование может подсчитываться в нескольких Счётчиках SAM.
- Результаты. Вкладка, на которой отображён чарт, визуализирующий полученные результаты подсчётов на временной шкале.
- Таблица результатов — список результатов подсчёта, в который добавляется новый результат подсчёта после каждого расчёта результатов Счётчика SAM. Результаты накапливаются в данной таблице для дальнейшего анализа и получения статистики.

Результат подсчёта является отдельной сущностью и содержит в себе основные поля результатов: Число прав, Число приложений, Лицензионный баланс, Консолидированный лиц. баланс, полученные на определённый момент расчёта.

- Консолидация. Вкладка, на которой происходит управление передачей прав между Счётчикам SAM.
- Счётчик предыдущей версии — Счётчик SAM, которому в случае отрицательного лицензионного баланса могут быть «переданы» лицензионные права с текущего счётчика (только если выбран признак Может передавать права). Указанный Счётчик SAM подсчитывает

лицензионное соответствие для того же программного обеспечения, но предыдущей по счёту версии.

- Может передавать права — признак, определяющий, что данный Счётчик SAM может передавать лицензионный права Счётчику предыдущей версии. Передача прав происходит, если в результате подсчёта у Счётчика предыдущей версии лицензионный баланс отрицателен, а у текущего счётчика — положителен, т.е. присутствует запас лицензионных прав, доступных к передаче. Лицензионные права могут передаваться по всей цепочке связанных Счётчиков SAM от самого верхнего уровня иерархии к самому нижнему уровню. Счётчик SAM может передавать не больше свободного количества лицензионных прав в его лицензионном балансе.
- Счётчик следующей версии — Счётчик, от которого в случае отрицательного лицензионного баланса могут быть «переданы» лицензионные права текущему счётчику (только если выбран признак Может передавать права). Указанный Счётчик SAM подсчитывает лицензионное соответствие для того же программного обеспечения, но следующей по счёту версии. Поле доступно только на чтение и является обратной ссылкой к Счётчику предыдущей версии.
- Детализация результатов. Вкладка, на которой возможно получить подробную «разбивку» полученных в полях Число прав и Число приложений значений с группировкой по указанным полям.
- Группировать лицензии по — полный путь к атрибуту, по которому будут детализированы результаты в поле Число прав. Поддерживается неограниченное число ссылок.
- Группировать приложения по — полный путь к атрибуту, по которому будут детализированы результаты в поле Число приложений. Поддерживается неограниченное число ссылок.
- История изменений. Список изменений атрибутов карточки.

9.8 Справочники

В данном разделе приведено описание общих Справочников в системе. Справочники представляют собой простые общие данные, используемые в других бизнес-сущностях.

В справочных сущностях присутствует общий признак Запись активна, который

определяет, используется ли данная запись в Системе на данный момент. Убрав указанный признак можно 'отключить' неиспользуемую/неактуальную запись, не удаляя её физически из таблицы.

Календари рабочих дней

Календарь — сущность, отражающая календарь рабочих дней в Системе. Календарь отражает рабочие, выходные, праздничные и сокращённые дни в году. Может использоваться для определения количества рабочих дней между двумя датами (например, при расчёте даты поставки), для определения количества рабочих часов для взятия задачи в работу (по SLA) и т.д.

Карточка календаря содержит информацию о типе года (високосный или невисокосный), а также об общем количестве дней и количестве выходных дней. На карточке присутствует визуализация календаря в виде специального компонента, где каждый отмеченный день является активным и кликабельным. При наведении курсора на день, рядом всплывает описание.

При нажатии на Нерабочий день, можно 'провалиться' в карточку данного дня.

При создании нового календаря, например на следующий год, нет необходимости

самостоятельно вводить все выходные и праздничные дни. При создании нового календаря нужно лишь сохранить вновь созданный Календарь, затем в карточке нажать на кнопку Синхронизировать и все данные по нерабочим дням за указанный год автоматически будут загружены в Систему.

Синхронизацию календаря лучше производить только по текущему или по следующему году, но в конце декабря.

Если выбрать Нерабочие дни, то раскрывшийся экран отобразит все дни, которые

полностью или частично не являются рабочими:

- Выходные;
- Праздничные;
- Сокращённые.

Помимо общепризнанных нерабочих дней, в Системе предусмотрено создание собственных дней, которые будут считаться нерабочими.

Для создания Нерабочего дня необходимо заполнить следующие поля:

- Ключ — автоматически генерируемое значение;
- День — необходимо указать дату для этого дня;
- Тип — необходимо указать один из трех возможных типов дней (см. выше):
- Описание — описание нового нерабочего дня, которое будет отображаться при наведении курсора.

При выполнении синхронизации на уже существующем календаре, обновятся только те дни, которые являются общероссийскими. Корпоративные нерабочие дни сохраняются.

Контрагенты

Контрагенты — сущность, определяющая, с какими компаниями текущая организация состоит в деловых отношениях. Контрагентом является лицо или учреждение, берущее на себя известные обязательства по договору.

Справочник контрагентов чаще всего используется в процессе управления соглашениями как неотъемлемый атрибут Договора и ему сопутствующих сущностей.

В карточке Контрагента должны быть доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Краткое название — сокращенное наименование контрагента;
 - Название — сокращенное наименование контрагента;
 - Работа в ЭДО — признак, определяющий, происходит ли работа с данным контрагентом в системе электронного документооборота;
- Контактная информация. Информация о координатах для связи с контрагентом:

телефон, email, страна, адрес и др.

Местоположения

Местоположение — сущность, определяющая, где/в каком месте расположен тот или иной объект. Местоположением может являться город, улица, дом, подъезд, этаж, помещение или стойка (определяется атрибутом Тип).

Является иерархическим справочником.

Справочник местоположений чаще всего применяется при управлении жизненным циклом актива как атрибут Актива, Сотрудника, Склада и др.

В карточке Местоположения должны быть доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Название — сокращенное наименование местоположения;
 - Тип — тип местоположения (город, улица, дом, подъезд, этаж, помещение или стойка);
- Основная информация. Информация об иерархии.
- Активы, размещённые здесь. Информация об Активах, размещённых в этом Местоположении. Одно Местоположение может содержать несколько Активов, а один Актив может быть расположен только в одном Местоположении.
- Склады. Информация о Складах, размещённых в этом Местоположении. Одно Местоположение может содержать несколько Складов, а один Склад может быть

расположен только в одном Местоположении.

Правообладатели

Правообладатель — сущность, определяющая компанию или юрлицо, которое имеет юридические права на продукт или наименование. Данный справочник является эталонным и содержит в себе уникальные названия на каждого учитываемого Правообладателя.

Справочник Правообладателей чаще всего применяется в Наименованиях, так как

определяет права на наименования ИТ-оборудования или лицензий.

В карточке Правообладателя доступны следующие вкладки и атрибуты:

- Шапка. Основные атрибуты:
 - Ключ — автоматически генерируемое значение;
 - Название — сокращенное наименование правообладателя;
- Правообладатели дискаверинга. Информация о Правообладателях дискаверинга, нормализованных к данному Правообладателю. К одному Правообладателю могут быть нормализованы несколько Правообладателей

дискаверинга, а один Правообладатель дискаверинга может быть нормализован только к одному Правообладателю. В данный список Правообладатели дискаверинга попадают в процессе нормализации данных.

- **Наименования.** Информация о Наименованиях, у которых указан данный Правообладатель. К одному Правообладателю могут относиться несколько Наименований, а одно Наименование может иметь только одного Правообладателя.

Наименования

Наименование — сущность, описывающая общие атрибуты тех или иных объектов. По своей сути является структурированным справочником-каталогом (или номенклатурой), содержащим в себе список ведущихся в компании названий и параметров материалов, лицензий, ИТ-оборудования и др. (любых сущностей, у которых указывается Наименование).

Так как Наименования всех сущностей физически находятся в одной таблице и

принадлежат к одной сущности Наименование (иногда также её называют Моделью), для разделения Наименований между собой (например, описывает ли данная запись наименование ПО или ИТ-оборудования) используется справочник Типов, на который ссылаются Наименования.

В разделе Наименования также присутствуют отдельные представления для каждой категории Наименований (подробнее ниже).

Наименования являются иерархическим справочником и могут описывать параметры разных категорий сущностей. Для каждой из данных категорий реализовано своё представление в меню и они разнесены по разным разделам в зависимости от принадлежности к какой-либо функциональной области.

Карточка Наименования меняется в зависимости от Типа. Некоторые вкладки видны для всех типов Наименований:

- **Шапка.** Общая информация: ключ, системное имя, парт номер, названия и др.
- **Общая информация.** Информация о типе, иерархии, правообладателе, единице измерения.
- **История изменений.** Список изменений атрибутов карточки.

Наименования приложений

Наименования приложений — представление списка Наименований, которые ведутся для сущности Приложения. Данный справочник является эталонным и содержит в себе уникальные названия на каждого учитываемого Наименования приложения.

В данном представлении отображаются только Наименования, у которых указан Тип 'Приложение' и в карточке Наименования помимо основных вкладок появляются следующие дополнительные:

- Приложения. Список Приложений, относящихся к данному Наименованию. С одним Наименованием может быть связано много Приложений, а Приложение может относиться лишь к одному Наименованию;

- Модели дискаверинга. Список Наименований дискаверинга, нормализованных к

данному эталонному Наименованию. К одному Наименованию может быть нормализовано много Наименований дискаверинга, а Наименование дискаверинга

может быть нормализовано лишь к одному Наименованию;

- Полномочия. Список Полномочий, связанных с данным Наименованием. К одному Наименованию может быть привязано много Полномочий, а Полномочие может быть привязано к нескольким Наименованиям. На данной вкладке происходит процесс управления полномочиями на установку и использование ПО.

Наименования ИТ-оборудования

Наименования ИТ-оборудования — представление списка Наименований, которые ведутся для сущности Активы с любым из типов ИТ-оборудования (компьютеры, СХД, сетевые устройства и др.).

В данном представлении отображаются только Наименования, у которых указан любой из дочерних Типов 'ИТ-оборудования' и в карточке Наименования помимо основных вкладок появляются следующие дополнительные:

- Активы. Список Активов, связанных с данным Наименованием. К одному Наименованию может быть привязано много Активов, а Актив может быть привязан лишь к одному Наименованию.

Наименования лицензий

Наименования лицензий — представление списка Наименований, которые ведутся для Лицензий ПО.

В карточке Наименования помимо основных вкладок появляются следующие дополнительные:

- Характеристики лицензии. Тип лицензии, число прав (данное значение учитывается в Счётчиках SAM), версия ПО (на которое выдаётся лицензия);
- Активы. Список Лицензий, связанных с данным Наименованием. К одному Наименованию может быть привязано много Лицензий, а Лицензия может быть привязана лишь к одному Наименованию.

Категории

Категории — представление списка Наименований, который предназначен для того, чтобы выдерживать иерархию в таблице Наименований и объединять другие Наименования в разделы для удобства восприятия.

В данном представлении отображаются только Наименования, у которых указан Тип 'Категория'. Во всех списках наименований Категории выделяются другим цветом.

Проекты

Проект — сущность, описывающая проект (цель, некая единица деятельности) Компании.

Является обычным справочником. Справочник проектов чаще всего применяется как атрибут Актива (в каком/для какого из проектов приобретался Актив).

Проекты добавляются и обновляются автоматически при синхронизации с Битрикс24, но их также можно создавать/редактировать вручную.

В карточке Проекта доступны следующие вкладки и атрибуты:

- Шапка.
 - Ключ — автоматически генерируемое значение;
 - Владелец — принадлежность данного Проекта к Владельцу (в случае, если настроено использование множества ЮрЛиц);
 - Запись активна — признак активности записи;
- Основная информация.

- Название — название проекта;
- Описание/цель — для чего создавался/проект;
- Ответственный — сотрудник Системы;
- Ключевые слова — тэги проекта;
- Код — код проекта;
- Дата начала — дата начала проекта;
- Дата окончания — дата окончания проекта;
- Bitrix id — числовое значение - id проекта в системе Битрикс24 (для связи);
- История изменений. Список изменений атрибутов карточки.

Склады

Склад — сущность, описывающая хранилище Активов в компании. Используется для отражения информации о том, где физически находится Актив. Является обычным справочником. Справочник складов чаще всего применяется как атрибут Актива, Перемещения и нужен для отражения фактического расположения Актива.

По умолчанию в Системе присутствует "абстрактный" Виртуальный склад НМА - место, в котором хранятся Лицензии (Активы типа Лицензия).

В карточке Склада доступны следующие вкладки и атрибуты:

- Шапка.
 - Ключ — автоматически генерируемое значение;
 - Название — наименование склада;
 - Запись активна — признак активности записи;
- Основная информация.
 - Местоположение — ссылка на справочник Местоположения;
 - Ответственный — сотрудник Системы;
- Размещенные активы. Здесь размещен список Активов, находящихся на этом Складе.
- История изменений. Список изменений атрибутов карточки.

Единицы измерения

Единица измерения — сущность, описывающая меру какой-либо характеристики в Системе.

Нужна для указания того, в каких единицах измеряются экземпляры сущностей. Является обычным справочником. Справочник единиц измерения чаще всего применяется как атрибут Наименования.

При развертывании Системы автоматически создаётся единица измерения Штука.

В карточке Единиц измерения доступна следующая вкладка и атрибуты:

- Шапка.
- Ключ — автоматически генерируемое значение;
- Полное название — полное наименование единицы измерения;
- Краткое название — наименование единицы измерения (кратко);
- Код — код единицы измерения (в верхнем регистре, для удобства);

10. ИНТЕГРАЦИИ

Система позволяет интегрироваться с различными сторонними системами любыми способами (REST, SOAP, подключения к БД и др.) Возможности интеграции ограничены только возможностями конкретной версии используемой JDK).

По умолчанию в Системе реализованы интеграции со следующими системами:

- LDAP;
- OCS Inventory (в качестве системы дискаверинга);
- Сайт-менеджер;
- Сайт ЦБ РФ;
- Битрикс24;
- Email.

В данном разделе описаны интеграции с различными сторонними системами.

LDAP

В Системе присутствует встроенный компонент (аддон) LDAP, который позволяет использовать возможности службы каталогов, например, Microsoft Active Directory. LDAP обеспечивает централизованное хранилище логинов и паролей, что позволяет приложениям подключаться к LDAP-серверу для подтверждения доступа.

С помощью интеграции с LDAP также происходит Синхронизация Пользователей.

При каждой авторизации будет происходить синхронизация пользователя Системы с пользователем LDAP.

Интеграция с Active Directory с использованием Jespa

В Систему также встроена поддержка расширенной интеграции со службой ActiveDirectory с использованием библиотеки Jespa по протоколу NTLMv2. Функциональность библиотеки позволяет браузерам получать информацию об авторизованном пользователе из Windows-сессии, за счет чего 'прокидывать пользователя' в Систему без необходимости указывать логин/пароль.

В случае использования аутентификации через Jespa синхронизация по LDAP работает только в случае явной авторизации через логин/пароль.

OCS inventory NG

В Системе по умолчанию настроена интеграция с системой инвентаризации и дискаверинга OCS Inventory NG (далее OCS).

OCS представляет из себя агентную систему сбора информации с удаленных устройств, и архитектурно является клиент-серверным приложением. Позволяет собирать и хранить информацию об аппаратном, программном обеспечении (с помощью программ-агентов, устанавливаемых на конечные устройства). Система использует данные, собранные с помощью OCS Inventory NG.

Сайт-менеджер

Интеграция с Сайт-менеджером (<https://sm.su/>) используется для получения информации о рабочих и нерабочих днях в Производственном календаре.

Для получения информации по дням текущего года выполняется GET-запрос REST API по адресу: <https://api.sm.su/v1/calendar/business/>

Выполняя этот запрос, Система получает ответ в виде JSON-файла с информацией по каждому дню года (рабочий/не рабочий, дата, тип, описание праздника и др.)

Сайт ЦБ РФ

Для получения курса Валют на указанную дату в системе используется интеграция с сайтом ЦБ РФ (<http://www.cbr.ru>)

Для получения информации по дням текущего года выполняется GET-запрос REST API по адресу: http://www.cbr.ru/scripts/XML_daily.asp

Выполняя этот запрос, Система получает ответ в виде XML-файла с информацией по курсам Валют на определённую дату.

Битрикс24

В Системе по умолчанию настроена функциональность синхронизации пользователей, их подразделений и проектов с CRM Битрикс24. Позволяет автоматически, с некоторой периодичностью, загружать и синхронизировать эти данные с Системой.

Email

Система предоставляет возможности синхронной и асинхронной отправки писем электронной почты из кода приложения. В этом разделе рассматриваются способы отправки, настройки SMTP-сервера и описание способов вызова отправки Email-писем из программного кода.

- Синхронная отправка — вызывающий код ожидает, пока сообщение не будет передано на SMTP-сервер;
- Асинхронная отправка — сформированное сообщение сохраняется в БД, управление сразу же передается вызывающему коду. Сообщение отправится позже из очереди отправки.