АСКОУ

Техническое описание

Version: 0.15

Last edited by: V.Titenko [tvolodi@abitech.kz](mailto:tvolodi@abitech.kz)

2022-09-23

Содержание

[1 Управление человеческими ресурсами (HR) 3](#_Toc111535547)

[1.1 Квалификация персонала 3](#_Toc111535548)

[1.1.1 Справочник квалификаций работников 3](#_Toc111535549)

[1.1.2 Справочник "Бригады" 3](#_Toc111535550)

[1.1.3 Справочник «Члены бригады» 3](#_Toc111535551)

[2 Техническое обслуживание и ремонт (ТоиР) (Maintenance and Repairs) 3](#_Toc111535552)

[2.1 Справочники 3](#_Toc111535553)

[2.1.1 Справочник статусов по статусу начала и окончания работы по наряд-заданию WorkStartFinishStatuses 3](#_Toc111535554)

[2.1.2 Справочник статусов наряд-задания по выполнению задач в срок или в нормы времени WorkNormStatuses 4](#_Toc111535555)

[2.2 Документы / операции 5](#_Toc111535556)

[2.2.1 Заявка/требование на регистрацию проблемы / дефекта / срока обслуживания 5](#_Toc111535557)

[2.3 Журналы 7](#_Toc111535558)

[2.3.1 Журнал запросов на обслуживание 7](#_Toc111535559)

[2.3.2 Журнал проведенных работ по обслуживанию оборудования 8](#_Toc111535560)

[2.4 Бизнес-процесс формирования заявки на обслуживание на основе периода 9](#_Toc111535561)

[2.5 Бизнес-процесс формирования заявки на обслуживание на основе показателей оборудования 9](#_Toc111535562)

[2.5.1 Заявка на обслуживание оборудования 10](#_Toc111535563)

[2.6 Наряд-задание на осмотр однотипного оборудования 12](#_Toc111535564)

[2.7 Простое наряд-задание 12](#_Toc111535565)

[2.7.1 Основные возможности 12](#_Toc111535566)

[2.7.2 Форма документа 13](#_Toc111535567)

[2.8 Повторяющееся простое наряд-задание 16](#_Toc111535568)

[2.9 Журналы 16](#_Toc111535569)

[2.9.1 Журнал наряд-заданий 16](#_Toc111535570)

[2.9.2 Журнал показаний работы оборудования (EntityReadings) 16](#_Toc111535571)

[2.10 План работы на период 16](#_Toc111535572)

[2.10.1 Генерация плановых наряд-заданий на период 17](#_Toc111535573)

[3 Internet of Things (IoT) / SCADA 17](#_Toc111535574)

[3.1 Общая схема 17](#_Toc111535575)

[3.2 Справочник контроллеров 18](#_Toc111535576)

[3.3 Логирование показаний датчиков 18](#_Toc111535577)

[4 Модуль "Трекинг объектов" (tracking) / Логистика / Транспорт 18](#_Toc111535578)

[4.1 Справочники 18](#_Toc111535579)

[4.1.1 Справочник "Маршруты" 18](#_Toc111535580)

[4.1.2 Справочник типов транспортных средств 18](#_Toc111535581)

[4.1.3 Справочник контейнеров 18](#_Toc111535582)

[4.1.4 Справочник «Транспортные средства» 19](#_Toc111535583)

[4.1.5 Справочник «Транспортные составы» 19](#_Toc111535584)

[4.1.6 Справочник маршрутов в разрезе транспортных средств 20](#_Toc111535585)

[4.1.7 Справочник типов пунктов маршрута 21](#_Toc111535586)

[4.1.8 Справочник пунктов маршрута 21](#_Toc111535587)

[4.1.9 Справочник звеньев маршрута 21](#_Toc111535588)

[4.2 Журналы 22](#_Toc111535589)

[4.2.1 Журнал регистрации местонахождений объектов 22](#_Toc111535590)

[4.2.2 Журнал регистрации выполнения объектом маршрута 22](#_Toc111535591)

[4.2.3 Журнал регистрации передвижения объекта по пунктам маршрута 23](#_Toc111535592)

[5 Модуль "Эконадзор" 23](#_Toc111535593)

[5.1 Справочник "Площадки контроля выбросов" 23](#_Toc111535594)

[5.2 Справочник "Источники выбросов" 23](#_Toc111535595)

[5.3 Справочник "Загрязняющие вещества" 24](#_Toc111535596)

[5.4 Справочник "Параметры загрязняющих веществ" 24](#_Toc111535597)

[5.5 Справочник "Нормативы допустимых выбросов" 25](#_Toc111535598)

[5.6 Журнал контроля выбросов для оператора №208 25](#_Toc111535599)

[5.7 Журнал контроля выбросов для инспекции №208 25](#_Toc111535600)

[5.8 Журнал превышения норм выбросов №208 / Уведомления о превышениях нормативов эмиссии по данным автоматизированной системы мониторинга 25](#_Toc111535601)

[6 Настройка рабочего процесса и отслеживания его выполнения 26](#_Toc111535602)

[6.1 Справочники 26](#_Toc111535603)

[6.1.1 Справочник типов документов DocTypes 26](#_Toc111535604)

[6.1.2 Справочник статусов наряд-задания DocStatuses 26](#_Toc111535605)

[6.1.3 Справочник вида маршрута документа WorkflowsTypes 26](#_Toc111535606)

[6.1.4 Определение списка статусов документа к переходу 27](#_Toc111535607)

[6.1.5 История движения документа по маршруту рабочего процесса 30](#_Toc111535608)

[6.2 Порядок работы со статусами документа при движении по маршруту 30](#_Toc111535609)

[6.3 Личный кабинет пользователя 30](#_Toc111535610)

[7 Бизнес-кейс: Отслеживание выполнения маршрута ж/д составами, учет пробегов и перевезенных грузов 30](#_Toc111535611)

[7.1.1 Определение пройденного расстояния 32](#_Toc111535612)

[7.2 Форма начального сопоставления локации и пробега 34](#_Toc111535613)

[7.3 Модуль трекинга объектов — идентификация вагонов и регистрация местонахождения оборудования 34](#_Toc111535614)

# Управление человеческими ресурсами (HR)

## Справочники

### Справочник квалификаций работников

Jira:

Для поддержания функциональности по работе с квалификациями работников в системе ведется справочник квалификаций следующей структуры:

Название справочника: Квалификации/Qualifications.

Модуль: Управление персоналом (HR).

Поля:

1. Код квалификации (необязательное) — строка / уникальные значения / необязательно.
2. Название квалификации — строка / обязательно.

### Справочник "Бригады"

Jira:

Для ведения планирования и учета работ в разрезе бригад используется справочник бригад:

**Название справочника**: Бригады / WorkTeams.

**Модуль**: Управление персоналом / HR.

**Поля**:

1. **Код бригады** — строка / уникальные значения(? Уникальные значения в пределах орг.единицы? ) / необязательно.
2. **Название бригады —** строка / обязательно.

### Справочник «Члены бригады»

Jira:

### Справочник «Организационные единицы»

# Техническое обслуживание и ремонт (ТоиР) (Maintenance and Repairs)

## Бизнес-процессы / Бизнес кейсы

### Общая схема формирования работ в модуле ТОиР

Процесс может иметь несколько начальных событий:

1. Результат измерения датчиками параметров / показателей оборудования и проверки этих параметров на соответствие нормам. Также такие параметры используются для определения необходимости проведения очередного технического обслуживания на основе пробега, моточасов. Или определения необходимости обслуживания на основе других правил или математической модели.
2. Результат осмотра оборудования специалистами, например, дефектоскопистом.
3. Результат ежедневного осмотра оборудования персоналом, который эксплуатирует это оборудование.
4. Результат наступления сроков планового обслуживания оборудования.

Все случаи необходимости обслуживания регистрируются в журнале запросов на обслуживание для централизованного учета таких потребностей.

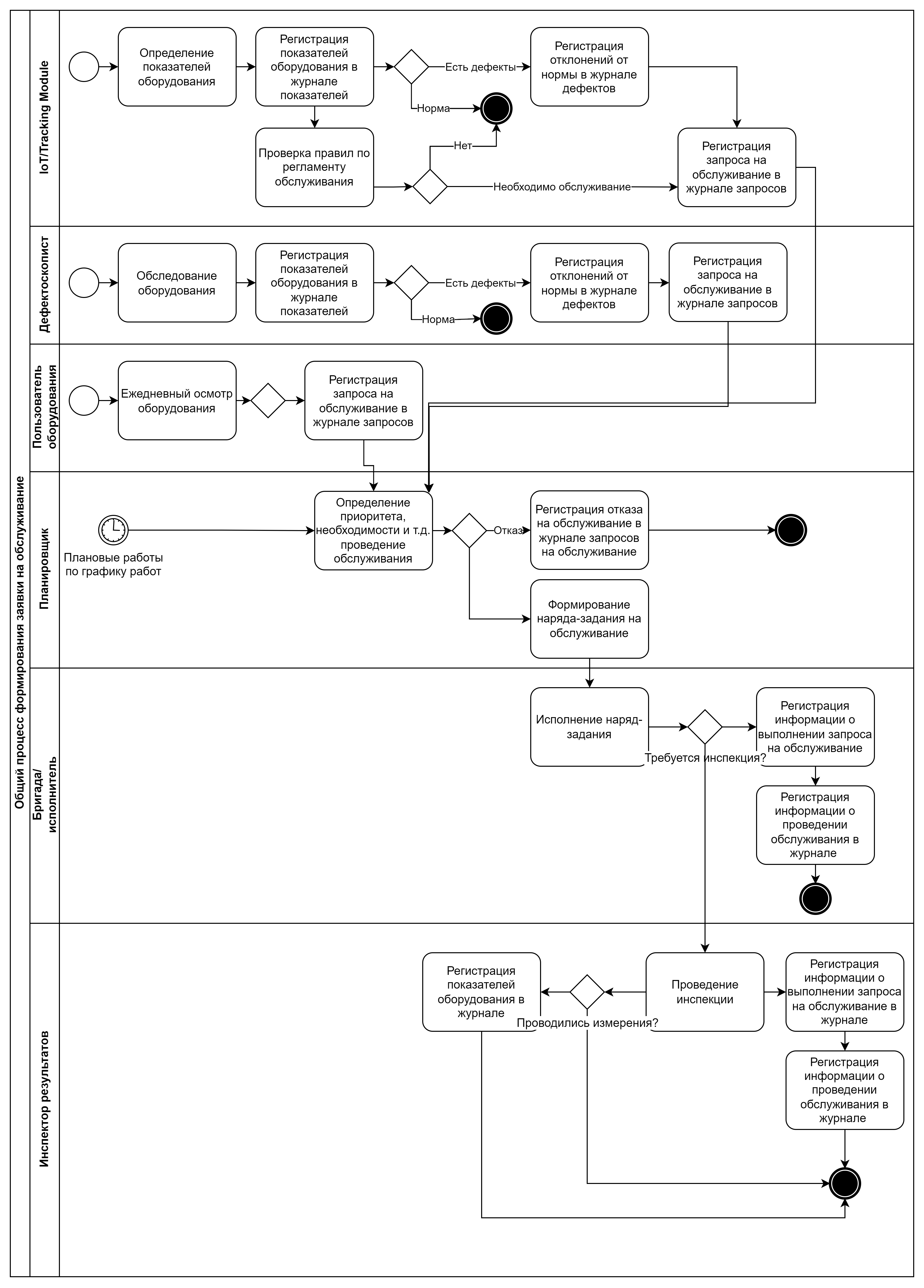
Запросы на обслуживание проходят через анализ планировщиком необходимости выполнения обслуживания. Планировщик на основе регламента обслуживания организации, приоритетов, ресурсов и т. д. определяет необходимость и возможность проведения обслуживания. В случае положительного решения формируется документ наряд-задания для выполнения такого обслуживания.

В случаях, когда регламентом предусмотрено автоматическое формирование наряд-задания, наряд-задание формируется системой на основе записи в журнале запросов на обслуживание.

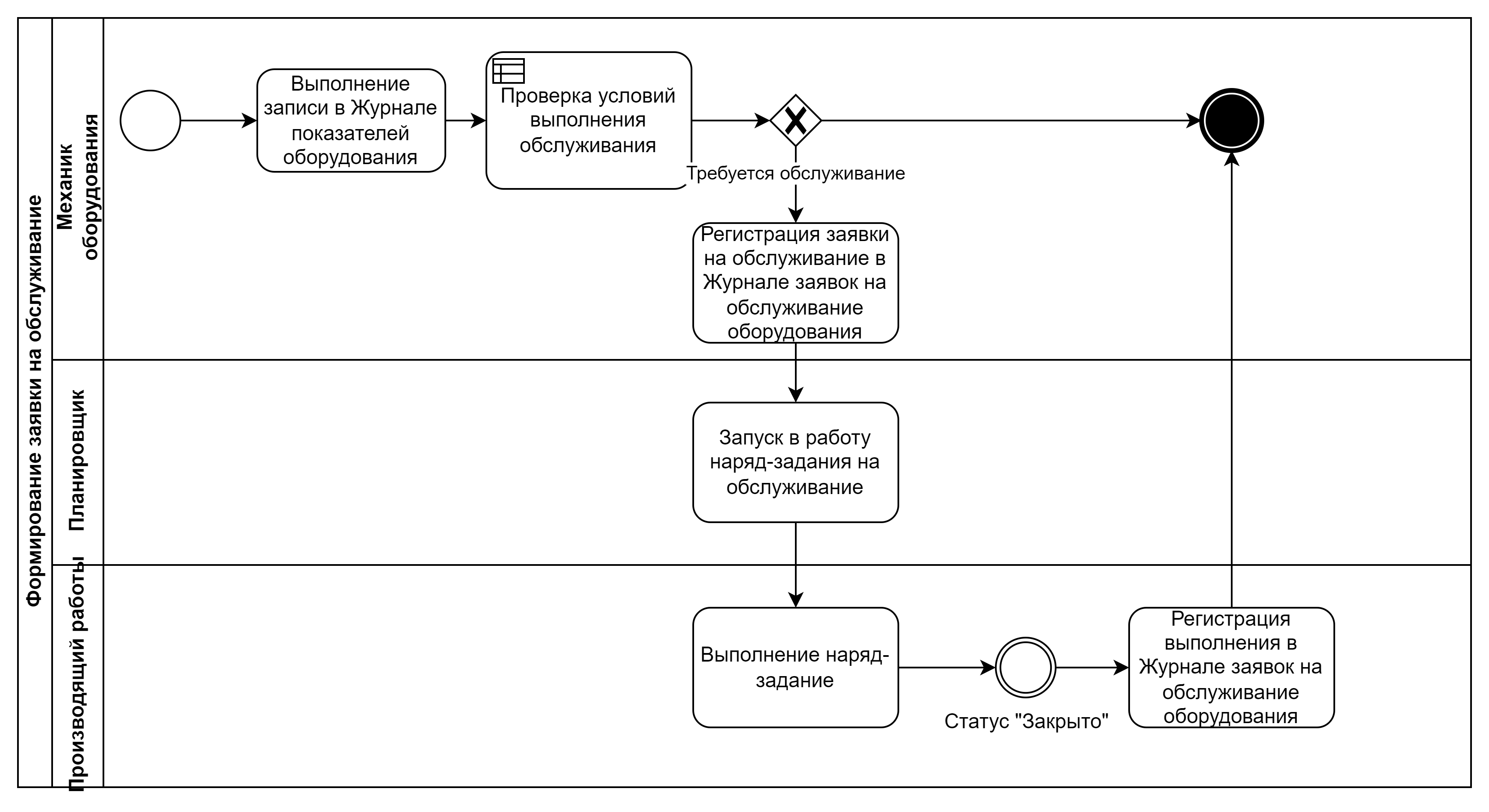
После окончания выполнения работ по обслуживанию (после закрытия наряд-задания) в журнале запросов на обслуживание проставляется информация о проведении обслуживания и закрытии, таким образом, запроса.

Последняя версия схемы бизнес-процесса:

[Схема бизнес-процесса](https://app.diagrams.net/" \l "Htvolodi-ab%2FASCOA%2Fmain%2Fmaint%2FРабочее задание. Процессы.drawio)



### Регистрация заявки на обслуживание

[](https://app.diagrams.net/" \l "Htvolodi%2Fabitech%2Fmain%2Fmaint%2FБП.Формирование заявки на обслуживание.drawio)Последняя версия схемы диаграммы

## Справочники

### Справочник статусов по статусу начала и окончания работы по наряд-заданию WorkStartFinishStatuses. AS-18

**Статус функциональности**: AB-74. Backlog.

<http://jira.abitech.kz:8080/browse/AS-18>

Справочник предназначен для информирования пользователя по статусу работы с точки зрения начала работы.

Структура справочника:

1. **Code** – строка – код записи. Обязательно к заполнению. Некоторые являются системными.
2. **Description** – строка – описание статуса. Обязательно к заполнению.

Перечень системных статусов:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| NOT\_STARTED | Не начата |
| IN\_TIME\_START | Начата раньше или в срок. Устанавливается, если фактическое время начала работ по наряд-заданию меньше или равно плановому времени. |
| LATE\_START | Начата позже. Устанавливается, если фактическое время начала работ по наряд-заданию больше планового времени завершения |
| IN\_TIME\_FINISH | Закончена раньше или в срок. Устанавливается, если фактическое время завершения работы меньше планового |
| LATE\_FINISH | Закончена с позже. Устанавливается, если фактическое время завершения работ больше, чем запланированное |

### Справочник статусов наряд-задания по выполнению задач в срок или в нормы времени WorkNormStatuses. AS-25

**Статус функциональности:**

[**https://ascoa.atlassian.net/browse/AB-75**](https://ascoa.atlassian.net/browse/AB-75)

[**http://jira.abitech.kz:8080/browse/AS-25**](http://jira.abitech.kz:8080/browse/AS-25)

Справочник предназначен для информирования пользователя по статусу работы с точки зрения выполнения в пределах периода времени.

Структура справочника:

1. **Code** – код записи. Обязательно к заполнению. Некоторые являются системными.
2. **Description** – описание статуса. Обязательно к завершению.

Перечень системных статусов:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| NOT\_STARTED | Не начата |
| AHEAD | Раньше срока |
| IN\_TIME | Вовремя |
| OVERDUE | Просрочено |

### Справочник нормативных значений показателей узлов. AS-14

Jira:

<http://jira.abitech.kz:8080/browse/AS-14>

Справочник предназначен для заданий нормативных показателей для узлов: геометрия, электрические, гидравлические и прочие.

Модуль: ТОиР.

Меню:

1. En: Maintenance > Directories > Equipment Node Standard Readings
2. Ru: ТОиР > Справочники > Нормативные показатели оборудования

Название справочника: **EqReadingsStandarts**.

Структура справочника:

1. **Id** – целое – заполняется на сервере.
2. **AssetId** – единица оборудования. Ссылка на справочник Активы / ОС / Оборудование. Необязательное поле к заполнению.
3. **AssetClassId** - ссылка на класс оборудования. **AssetClass** модуля **Inventory**. Обязательно к заполнению, если не заполнено поле **AssetId**.
4. **ReadingsTypeId** – тип показателя. Ссылка на справочник типов показателей.
5. **ComparisonType** - Тип сравнения (server/InfrastructureApp.Base/Enums/Regulations/ComparisonType.cs) :
   1. Равно - =
   2. Меньше - <
   3. Больше - >
   4. Меньше или равно - ≤
   5. Больше или равно - ≥
   6. Между – между/between
   7. Вне – вне/out
6. **MinValue** – вещественный тип.
7. **MaxValue** – вещественный тип.

Функциональность справочника:

1. Добавить/удалить/отредактировать.
2. Выбор из drop-down - достаточно простого выбора из списка, без отдельной экранной формы выбора.

### Справочник «Ремонтные подразделения»

Jira:

<http://jira.abitech.kz:8080/browse/AS-104>

Справочник предназначен для учета подразделений, которые могут проводить ремонты. Участвует в выборках оргединиц по обслуживанию и формированию отчетов в разрезе ремонтных подразделений.

Название справочника: **RepairOrgUnits**. Модуль **Infrastructure**.

Структура справочника:

1. Идентификатор – целое – заполняется в режиме онлайн.
2. Подразделение компании – ссылка на справочник оргединиц модуля HR.

Функциональность: стандартная.

Отображать название подразделения с учетом всей иерархии через /. Например: Суперкомпания/Филиал 1/Ремонтный цех 123.

### Справочник «Типы дефектов оборудования» (AS-107)

Jira:

<http://jira.abitech.kz:8080/browse/AS-107>

Справочник предназначен для систематизации типовых дефектов оборудования по классу оборудования.

Название таблицы: **EqDefectTypes**. Модуль **infrastructure**.

Структура:

1. **Id** – Guid – типы могут создаваться в на рабочем месте в режиме офлайн (?).
2. **Code** – string – код. Может быть пустым. Уникальное значение.
3. **AssetClassId** – класс оборудования - ссылка на справочник классов оборудования / основных средств (модуль **inventory**). Может быть пустым – общий тип.
4. **Description** – string – описание типа.
5. **ExtDescription** – string – расширенное описание.

Функциональность – стандартная.

## Документы / операции

### Заявка/требование на регистрацию проблемы / дефекта / срока обслуживания

Документ предназначен для формирования заявления/требования, когда заявитель считает, что оборудование нуждается в проведении обслуживания, ремонта, осмотра и т. д. Также документ создается, когда наступает срок проведения такого обслуживания. Запись может создаваться как по инициативе оператора, так и на основе логики системы.

Модуль: **ТОиР**.

Таблица в базе данных: **MaintReqests**.

Меню:

**Структура документа**:

1. **Id** – GUID. Записи могут создаваться на мобильном клиенте.
2. **DocNumber** – строка — номер документа.
3. **DocDateTime** – DateTimeOffset – дата и время документа. Текущее дата и время. Нельзя редактировать.
4. **Description** – описание документа.
5. **DocTypeId –** ссылка на тип документа модуля **Workflow**. Ссылка на запись с кодом «**MaintenanceRequest”**. Запись необходимо завести в справочнике **DocTypes** модуля **Workflow**.
6. **WorkflowTypeId** – ссылка на таблицу рабочих процессов.
7. **WorkflowStatusId** – статус документа в движении по рабочему маршруту. Ссылка на справочник статусов рабочего процесса (модуль Workflow).
8. **EqId** – ссылка на справочник «Основные средства / Оборудование» (Assets).
9. **MaintenanceTypeId** – ссылка на тип обслуживания.
10. **RequestReason** – строка — причина для проведения обслуживания.
11. **SourceDocTypeId** – строка — тип документа источника — документа, на основе которого создан данный документ. Ссылка на справочник **DocTypes** / модуль **Workflow**.
12. **SourceDocRefId** – строка — ключ записи — документа источника.
13. **ResultDocTypeId** – строка — тип дочернего документа, который был порожден данным документом. Ссылка на справочник **DocTypes** / модуль **Workflow**.
14. **ResultDocId** – строка — ключ записи — дочернего документа.

**Форма документа**:

**Шапка документа**:

Меню «Дополнительные операции»:

* Сформировать дефектную ведомость.
* Сформировать наряд-задание.

**Вкладка «Основания для обслуживания»:**

Содержит табличную часть детализация причин для обслуживания и предназначена для расшифровки причин для проведения обслуживания, если их несколько:

Таблица в базе данных: **MaintRequestReasons.**

Структура:

1. **Id** – Guid – может заполняться на мобильном клиенте.
2. **MaintRequestId** – ссылка на запрос на обслуживание.
3. **Description** – строка — описание причины. Обязательное для заполнения.

**Вкладка «Запланированное обслуживание по ед.оборудования»:**

Вкладка предназначена для формирования предварительного

1. № п/п.
2. **MaintenanceTypeId -** тип обслуживания. Необязательное поле. Ссылка на справочник «**Типы обслуживания**».
3. **PlannedReadings** - плановый показатель (пробег, выработка, кроме даты). Ссылка на Справочник «**Типы показателей»**. Необязательное поле.
4. **Readings** - Текущий показатель. Обязательное поле, если заполнено поле «**Плановый показатель**».
5. **UoMId -** единица измерения. Обязательное, если заполнено поле «**Плановый показатель**». Ссылка на справочник «**Единицы измерения**».
6. Плановая дата (сравнивается с датой документа). Необязательное поле.

Должно быть заполнено или поле **Плановый показатель** или **Плановая дата**.

**Вкладка - Табличная часть «Проведенные работы с единицей оборудования»**

Табличная часть является информационной и предназначена для вывода информации по уже проведенным работам по обслуживанию этой единицы оборудования из **Журнала проведенного обслуживания**. Заполнение данными проводится после заполнения поля «**Единица оборудования**».

Состав:

1. № п/п.
2. Тип обслуживания - …
3. Дата обслуживания -
4. Ответственный -

Вкладка - Табличная часть «Текущие показатели»

1. № п/п.
2. Тип показателя.
3. Величина
4. Ед. измерения.

## Наряд-задание

Наряд-задание – документ, который описывает плановые характеристики работы, фиксирует ход выполнения работы, а также результат.

**Модуль**: ТОиР.

**Меню**: ТОиР – Наряд-задание комплексное (en: Complex work-order)

**Основная таблица**: Jobs.

### Общая информация

http://wiki.askou.kz/?page\_id=1821

Наряд задание представляет собой сложный документ, работа с которым предполагает начало работы с простой формы с постепенным добавлением необходимых компонентов документа по мере необходимости. Наряд-задание позволяет выполнять следующие функции:

1. Описывать требуемую работу.
2. Ставить нормативные временные рамки выполнения работы и фиксировать фактические.
3. Детализировать работу путем указания детальных задач.
4. Формировать маршрут из единиц оборудования, на каждой из которых необходимо выполнить требуемые работы.
5. Фиксировать состояние оборудования с использованием фотоснимков и фиксацией показателей оборудования.
6. Привязывать работу к организационной структуре предприятия и месту выполнения работ.
7. Приоритезировать работы.
8. Регистрировать исполнителей по работе.
9. Регистрировать требуемые материалы и специальное оборудование для выполнения работы.
10. Привязывать инструкции, приказы, распоряжения, другие документы, имеющие отношение к работе.
11. Фиксировать посменную работу.
12. Работа в паре с документом «Наряд-задание».
13. Документ имеет свой жизненный цикл, т. е. проходит через несколько статусов, которые отражают важные вехи в процессе выполнения или организации работы. На каждом этапе с документом работают другие пользователи с соответствующими доступами к функциональности документа.

Группы информации:

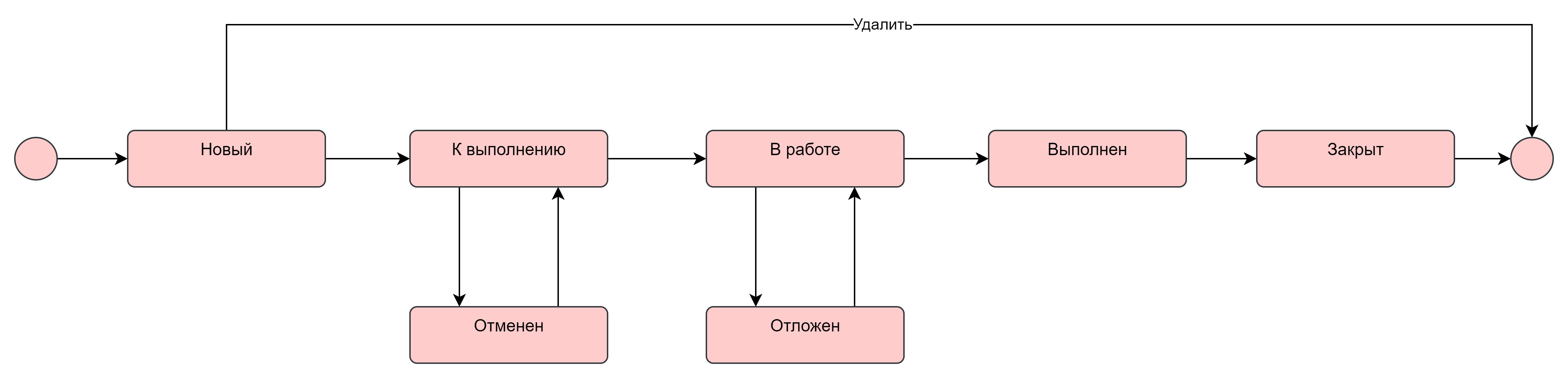
1. Общая информация.
   1. Нумерация документов.
   2. Дата документа / операции.
   3. Приоритет.
   4. Планировщик.
   5. Исполнитель / прораб / бригадир / …
   6. Статус документа – состояние операции.
2. Плановые даты и периоды:
   1. Плановая дата начала работы.
   2. Плановая дата окончания работы.
   3. Плановая дата останова оборудования.
   4. Плановая дата запуска оборудования.
3. Фактические даты и периоды:
   1. Фактическая дата начала работы.
   2. Фактическая дата окончания работы.
   3. Фактическая дата останова оборудования.
   4. Фактическая дата запуска оборудования.
4. Нормы:
   1. Затрат времени на операции.
   2. Затрат материалов на операцию.
   3. Затрат материалов на узел.
   4. Затрат труда на операцию.
   5. Затрат труда на узел.
   6. Затрат специального оборудования на операцию.
   7. Затрат специального оборудования на узел.
5. Квалификационные требования к выполнению работ.
6. Допуск к работе по прохождению инструктажей.

История:

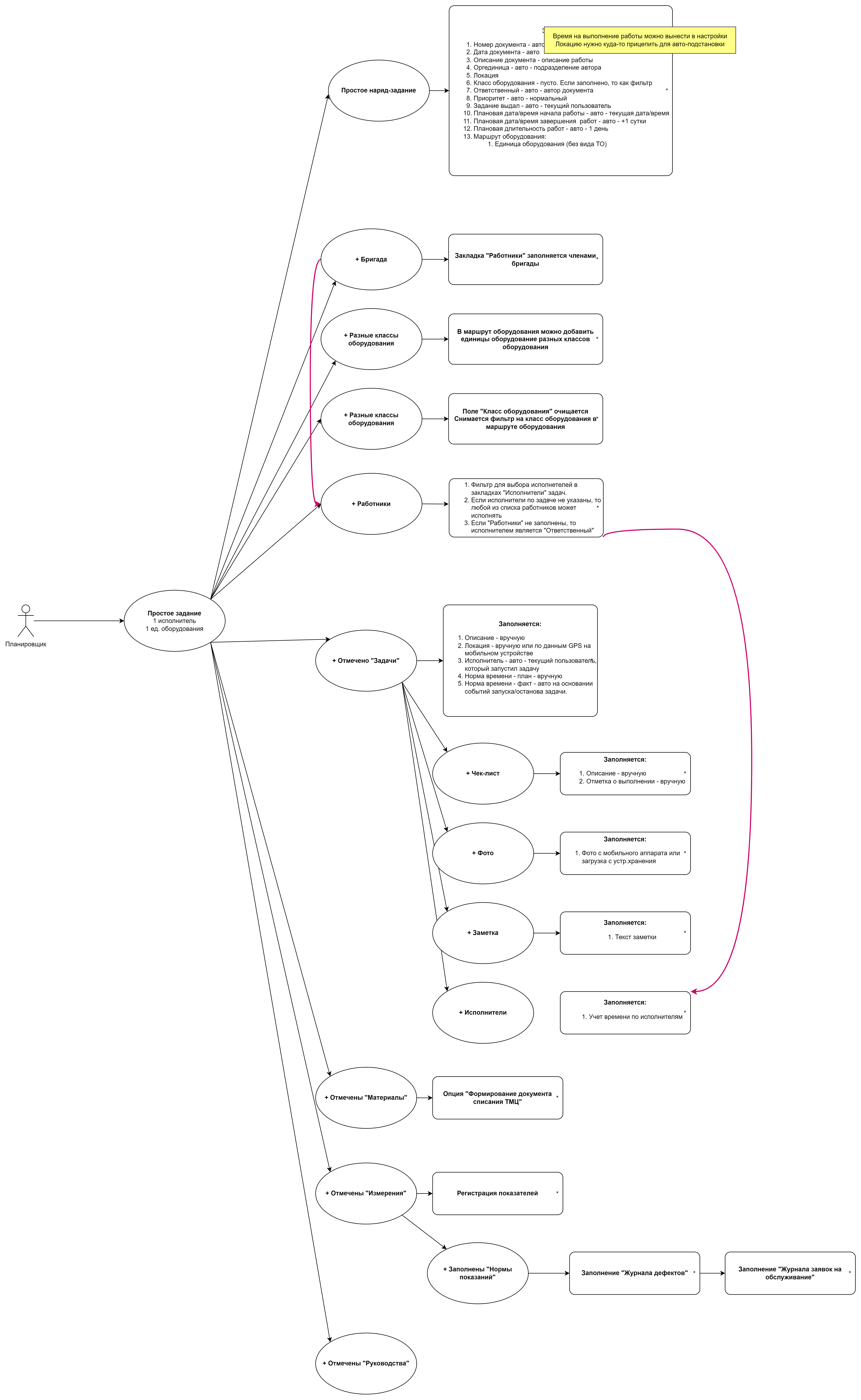
1. Выполнения операций: переходы по статусам документа с регистрацией исполнителя и времени.
2. Скорость прохождения документа по рабочему маршруту.

Основные статусы документа:

1. **Новый/черновик/draft** – присваивается документу при создании. Основное назначение статуса — подготовка данных по работе к выполнению.
2. **К выполнению/To Execute** – присваивается документу планировщиком или лицом, которое уполномочено выдавать работы. Основное назначение статуса — отметка работы как спланированной к выполнению, а также сигнал о том, что эта работа должна быть выполнена.
3. **В работе/In Progress** – присваивается документу, когда работа по документу начала и продолжает выполняться. В этом статусе регистрируются параметры выполнения работы.
4. **Выполнен / Done** – присваивается документу, когда работа по документу завершена.
5. **Закрыт / Closed** – присваивается документу, когда документ завершен и в дальнейшем он используется только для информационных целей.



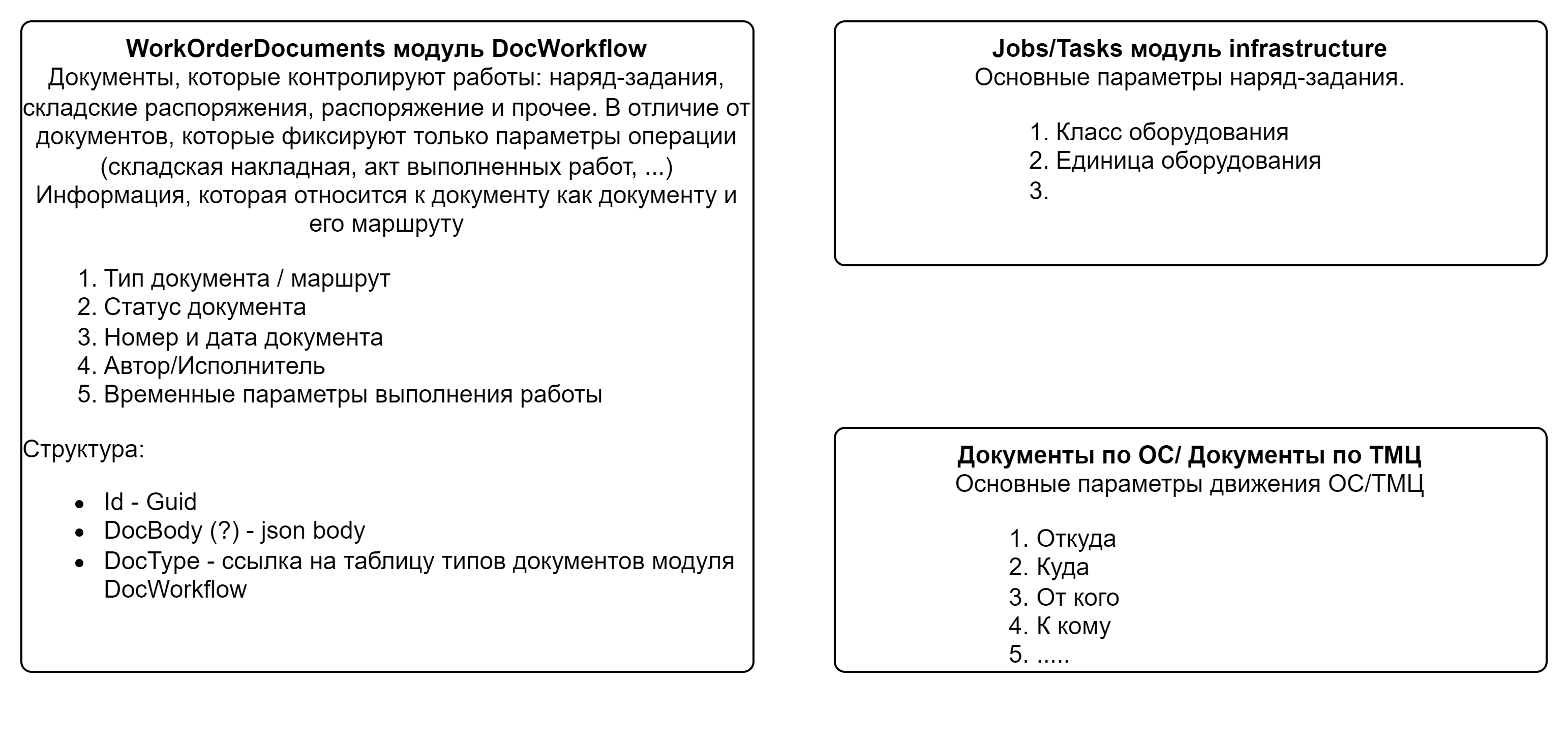
### Функции документа. Схема



### Способ хранения данных документа

Наряд-задание является распорядительным документом, который описывает и регистрирует операции при выполнении работ. Данные документа хранятся в таблицах:

1. **WorkOrderDocuments** модуля **DocWorkflow –** данные, которые относятся к документу как документу и его движению по рабочему процессу. Характеристики операций с точки зрения затрат времени и ресурсов по работе вцелом.
2. **Jobs/Tasks модуля Infrastructure** – данные, которые относятся к работе с оборудованием и задач работы. Это класс оборудования, единица оборудования, перечень задач, учет времени выполнения задач, которые являются скорее технологическими операциями, прочее.



### Форма документа

Форма может использоваться как для простых работ, так и для сложных. Пользователь может добавлять дополнительные характеристики путем проставления отметок на нужных компонентах формы.

[Последняя версия схемы формы](https://app.diagrams.net/" \l "Htvolodi-ab%2FASCOA%2Fmain%2Fmaint%2FРабочее задание.drawio)

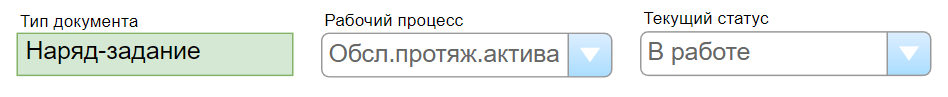
Основные статусы документа:

1. Компонент данных по работе, состоящая из нескольких закладок, которые содержат разные классы атрибутов документа: данные по работе, данные по связанным документам, подписи и прочее.
2. Секция регистрации данных по маршруту документов и связанным операциям – перечень оборудования и связанных с ним работ.

#### Компонент “Рабочий процесс документа». AS-39

Jira:

<http://jira.abitech.kz:8080/browse/AS-39>



Компонент используется для:

1. Определения рабочего процесса, по которому должен двигаться документ.
2. Определения текущего статуса документа.

Состав компоненты:

1. **Тип документа рабочего процесса**. Всегда – «**Наряд-задание**». Регистрируется в типах документов модуля рабочего процесса (Workflow). Поле изменить нельзя.
2. **Рабочий процесс** – ссылка на тип рабочего процесса модуля рабочего процесса.
   1. При создании нового документа поле остается пустым.
   2. Пользователь выбирает, по какому рабочему процессу будет двигаться документ. Если в справочнике типов документа указан рабочий процесс по умолчанию, то проставляется указанный тип рабочего процесса.
   3. Поле можно поменять только в статусе документа «**Черновик**».
3. **Текущий статус** – ссылка на статус рабочего процесса. Показывает на каком шагу рабочего процесса находится операция и соответственно документ.
   1. Пользователь может перевести документ из текущего статуса в другой, выбрав следующий статус и сохранив документ.
   2. Чтобы пользователь мог перевести документ в следующий статус, у него должны быть соответствующие полномочия. Более подробно см. 6.1.4 .
   3. Событие изменения статуса документа регистрируется в журнале изменений статусов документов. См. здесь: 6.1.5

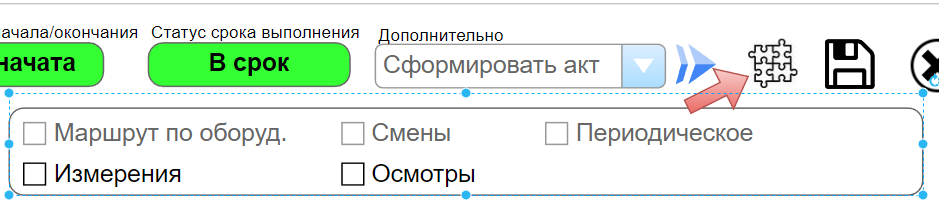
Вышеперечисленные поля хранятся в таблице **WorkOrderDocuments** модуля **infrastracture**.

1. **Статус начала/окончания работ** – отображает статус работы с точки зрения начала и завершения работы.
2. **Статус документа с точки зрения выполнения работы в срок**.
3. **Дополнительные действия с документом** – меню, из которого можно выбрать дополнительные действия и выполнить. Например, импортировать документ, экспортировать, распечатать, создать новый на основе текущего, создать акт выполненных работ, другое.

#### Компонент «Дополнительные компоненты документа» AS-102

Jira:

***http://jira.abitech.kz:8080/browse/AS-102***



При создании документа его форма имеет только минимальный набор атрибутов для работы.

Если пользователю необходимы дополнительные функции формы, то он их может подключить через «**Дополнительные компоненты**». Для этого используется FieldSet или Advanced Panel (PrimeVue) c набором чек-боксов для включения/выключения соответствующего компонента.

Набор чек-боксов хранится в таблице **DocComponents** модуля **DocWorkflow**. Таблица имеет следующие поля:

1. **Id** – Guid – может создаваться офлайн.
2. **DocId** – ссылка на документ.
3. **ComponentId** – компонент для наряд-задания. Ссылка на справочник компонентов документа в модуле **DocWorkflow** - **DocComponentTypes**.
4. **ComponentIsOn** – булево значение — признак, что компонент включен (или нет).

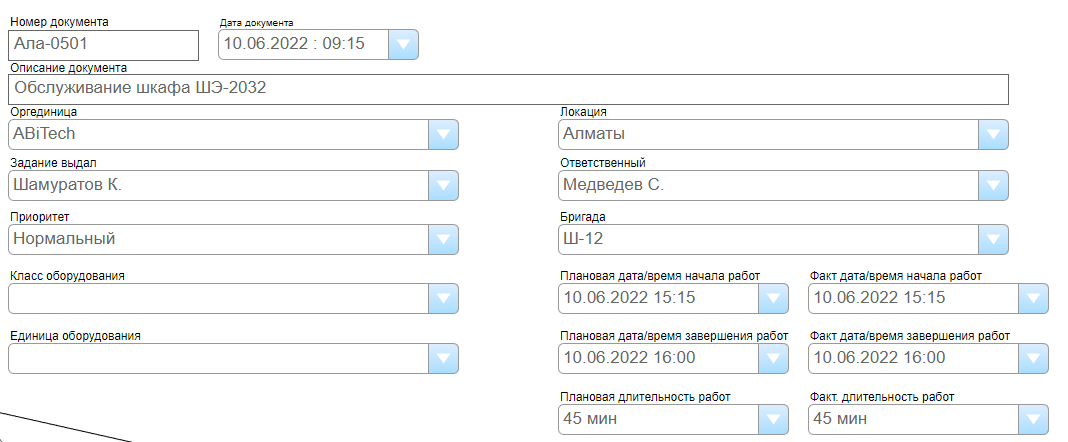
Возможные действия:

1. Открыть/закрыть панель включения компонент.
2. Выбрать нужные компоненты.
3. При построении перечня переключателей для компонент система проверяет их доступность по таблице **DocTypeComponents** в модуле **DocWorkflow.**
4. При включении компонента формы на экране появляются дополнительные элементы управления (см. соответствующий компонент).
5. При сохранении документа настройки компонентов сохраняются в таблице.
6. При открытии документа проверяется, что уже есть настройки компонентов. Если есть, то активные компоненты отображаются.

#### Компонент основных данных по документу. AS-41

Jira:

<http://jira.abitech.kz:8080/browse/AS-41>



Компонент предназначен регистрации в БД основных данных документа. Данный компонент не может быть выключен.

1. **DocNumber – Номер документа** — генерируется системой на основе счетчика, который сбрасывается в начале года. Обязательное поле.
2. **DocDate – Дата документа** — текущая дата и время документа.
3. **Description – Описание документа** — краткое описание сути наряд-задания. Опциональное поле.
4. **OrgUnitId – Оргединица** — подразделение, где проводится обслуживание.
5. **PriorityId – Приоритет** — выбор из справочника приоритетов.
6. **WorkIssuerUserId – Задание выдал** – ссылка на работника, кто перевел документ в состояние «**Запланировано**»/«**К исполнению**».
7. **WorkResponsibleUserId – Производящий работы** — ответственный/бригадир за непосредственное выполнение работ. Заполняется работником - планировщиком или системой тем работником, кто перевел документ в состояние «**В работе**».
8. **WorkTeamId – Бригада** — выбирается из справочника «**Бригады**». Заполняется, если работа производится одной бригадой.
9. **LocationId – Локация** — выбирается из справочника «**Локации**». Общее место проведение работ, место проведения отдельных операций может уточняться в секции регистрации хода работ.
10. **ScheduledOn – Плановая дата/время начала работ**: заполняется планировщиком.
11. **ScheduledFinishOn – Плановая дата/время окончания работ**: заполняется планировщиком.
12. **Плановая длительность работы**: заполняется планировщиком.
13. **StartedOn – Фактическая дата/время начала работ**.
14. **FinishedOn – Фактическая дата/время окончания работ.**
15. **Фактическая длительность работ -** разница окончания последней работы и начала первой работы.

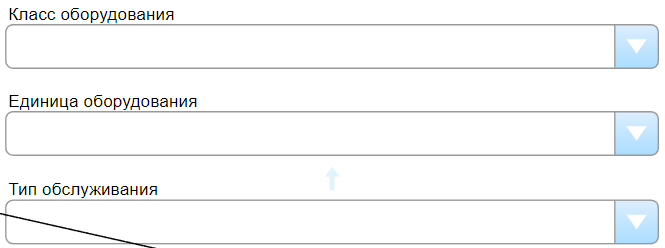
Основные данные хранятся в основной таблице документов заданий на работы **WorkOrderDocuments** модуля **DocWorkflow**.

Заполнение полей компонента:

* 1. **Номер документа** - заполняется автоматически. Пользователь может поменять. Комбинация «Номер документа» + «Дата документа» является уникальной.
  2. **Дата документа** - заполняется текущей датой + текущее время. Пользователь может сам выставить, используя элементом управления «Календарь». Обязательное поле. Можно изменять, пока документ в состоянии «**Черновик**».
  3. **Оргединица** - подразделение, в котором работает автор документа. Пользователь может выбрать другую, доступную ему оргединицу. Обязательное поле.
  4. **Приоритет** — значение с кодом «50». Проследить, что в таблице приоритетов есть такая запись.
  5. **Задание выдал** – ссылка на работника, кто перевел документ из состояния «**Черновик**» в следующий статус.
  6. **Производящий работы** — заполняется работником-планировщиком или системой тем работником, кто перевел документ в состояние «**В работе**».
  7. **Фактическая дата/время начала работ**: заполняется производящим работы или автоматически при начале первой работы в списке работ.
  8. **Фактическая дата/время окончания работ**: заполняется производящим работы или автоматически при завершении последней работы в списке работ.
  9. **Фактическая длительность работ**: заполняется производящим работы или автоматически как разница окончания последней работы и начала первой работы.

#### Компонент «Данные по обслуживанию единицы оборудования»

Jira:



Компонент заполняется для привязки работы к классу оборудования и единице оборудования.

Компонент заполняется, если не выбран компонент «**Маршрут по оборудованию**».

1. **Класс оборудования —** выбор из справочника «**Классы оборудования / ОС**». Обязательное поле, если в опциях компонентов документа не указано «**Маршрут по оборудованию**».
2. **Единица оборудования** — выбор из справочника «**Оборудование / ОС**». Обязательно поле, если в опциях документа не выбрано «**Маршрут по оборудованию**».
3. **Тип обслуживания** – выбор из справочника «**Типы обслуживания**». Необязательное поле.

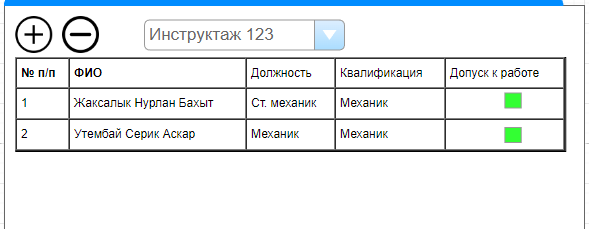
Заполнение:

1. Если указан «**Класс оборудования**», то оборудование нужно фильтровать по этому классу.
2. Если «**Класс оборудования**» не заполнил, то при выборе единицы оборудования заполнять класс оборудования из единицы оборудования.

Данные хранить в таблице **Jobs** модуля **Infrastructure**. В записи таблицы **Jobs** предусмотреть поле **WorkOrderDocumentId**, в котором регистрировать ссылку на запись документа в таблице **WorkOrderDocuments**.

#### Компонент «Работники»

Jira:



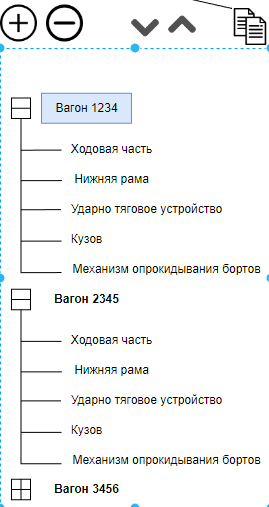
Компонент предназначен для учета работников, которые назначены и допущены к работе. Представляет собой форму-закладку с табличной частью. Записи в табличной части можно добавлять и удалять. Поля табличной части:

1. **№ п/п** – порядковый номер строки.
2. **ФИО, Должность, Квалификация** – заполняются данными выбранного работника.
3. **Допуск к работе** – заполняется на основе записи в журнале инструктажа. Для допуска к работе проводится инструктаж в **ОТ и ТБ**. Далее этот инструктаж выбирается в меню компонента. Если работник найден в записях инструктажа в журнале инструктажа, то ему проставляется «**Допуск к работе**». Бригадир может проставить допуск к работе, если поле-ссылка на инструктаж пустое.

При переводе документа в статус «**В работе**» данные компонента сохраняются в **Журнале допусков к работам по оборудованию** в модуле infrastructure.

#### Компонент «Маршрут оборудования»

Jira:



Компонент предназначен для построения маршрута работы с оборудованием в случае, когда работа затрагивает несколько единиц оборудования, при этом работы нужно провести по каждой единице. Компонент отображает список из единиц оборудования, дочерние узлы которых отображаются в виде дерева. К позициям дерева узлов привязан блок компонентов, которые описывают работы с этим узлом, необходимые для этого ресурсы, а также информацию по состоянию узла.

Заполнение маршрута может проводиться автоматически на основе других документов или событий в системе, вручную, в том числе способом считывания метки на единице оборудования.

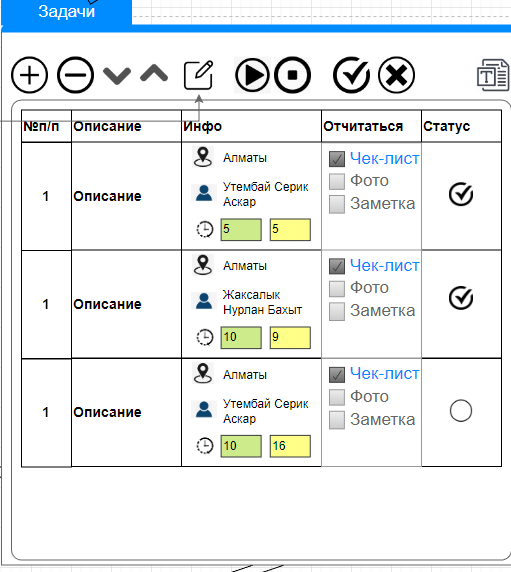
Действия:

* 1. Добавить единицу оборудования в дерево. Если у единицы оборудования есть дочерние узлы, то они добавляются тоже. Можно выбрать, должны ли добавляться дочерние узлы. Добавление выполняется только в корень дерева.
  2. Удалить единицу оборудования из структуры дерева.
  3. Поменять позицию единицы оборудования в списке оборудования, переместив вверх или вниз.
  4. Скопировать настройки работ с текущего узла на **все** остальные. Можно выделить нужные узлы, тогда копирование выполнится только на выделенные узлы. Копируются:
     1. Описания задач.
     2. Локации задач.
     3. Нормы времени на задачу.
     4. Настройки по отчетности по задаче.
     5. Шаблон чек-листа.
     6. Список исполнителей по задаче.

Данные маршрута хранятся в **OrderDocuments** модуля **DocWorkflow**.

#### Компонент «Задачи по оборудованию»

Jira:



Компонент предназначен для детализации задач, которые необходимо выполнить по единице оборудованию, а также для контроля выполнения этих задач. Перечень оборудования для каждой единицы оборудования свой. Если необходимо выполнить одни и те же задачи по всем единицам оборудования, то описываются задачи для одной единицы, после чего они копируются на остальные единицы (см. компонент «**Маршрут по оборудованию**»).

Колонки табличной части:

1. № п/п — номер по порядку.
2. Описание содержание работы.
3. **Инфо** — колонка с комбинированным набором элементов. Где, кто, как долго.
   1. **LocationId** – Локация — выбирается из справочника локаций — **hr.Locations**.
   2. Последний исполнитель по этой задаче — ссылка на справочник **hr.Users**. Берется последний по этой задаче по времени записи в таблице **TaskExecEmployees** – реестр работников-исполнителей по задаче. Если такой записи нет, то берется первый исполнитель из закрепленных работников за задачей из реестра **TaskEmployees** по этой задаче. Если таких тоже нет, то оставляется пустым.
   3. **StdDurationTicks** – длинное целое – назначенное время на выполнение работы (сохраняется в тиках времени).
   4. **DurationTicks** – фактически потраченное время на выполнение работы (сохраняется в тиках времени).
4. **Отчитаться** — набор признаков необходимости отчитаться определенным образом о результатах выполнения работы.
   1. **IsAttachmentRequired** – признак, что для завершения задачи необходимо прикрепить дополнительный файл.
   2. **IsEquipmentNotesRequired** – признак, что для завершения задачи необходимо заполнить примечание к задаче.
   3. **IsCheckListRequired** – признак, что для завершения задачи необходимо выполнить операции по чек-списку.
   4. **IsReadingsRequired** – признак, что для завершения задачи необходимо провести измерения параметров единицы оборудования.
   5. Время: норма/факт. Если поле «**Факт**» превышает «**Норма**», то оно окрашивается в желтый цвет.
   6. Время начала/окончания задачи. На экране не отображается (пока). Детальный учет ведется в таблице **infrastructure.TaskStates**.

Вышеперечисленные данные при сохранении документа регистрируются в журнале задач: **infrastructure.Tasks**.

1. **Статус** — Текущий статус задачи. Для статусов «**Выполнено**», «**Не начато**», «**На паузе**» - свои иконки. Регистрируется при сохранении в таблице **infrastructure.TaskStates**.

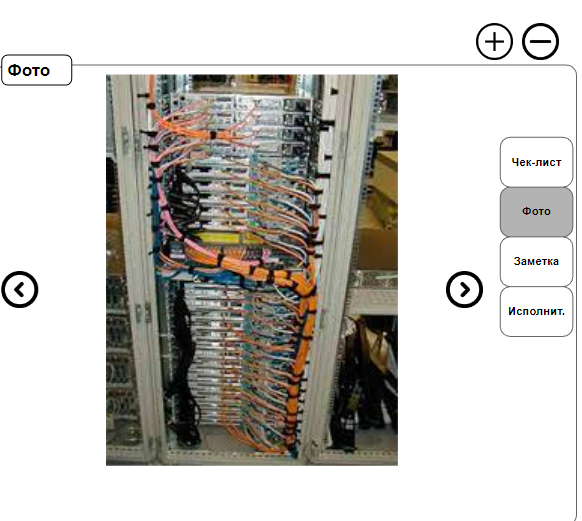
Панель инструментов для табличной части «**Задачи**» включает:

1. Кнопка «**Добавить**» задачу — добавить новую строку в таблице.
2. Кнопка «**Удалить**» задачу — удалить строку задачи.
3. Кнопка «**Поднять вверх**» задачу в списке.
4. Кнопка «**Опустить вниз**» задачу в списке.
5. Кнопка «**Редактировать**». Переводит выделенную строку задач в режим редактирования.
6. Кнопка «**Начать/продолжить задачу**». При нажатии на кнопку в таблице **infrastructure.TaskStates** формируется запись для задачи со статусом **Started**.
7. Кнопка «**Остановить**» выполнение задачи — в **infrastructure.TaskStates** формируется запись для задачи со статусом **Paused**.
8. Кнопка «**Выполнено**» для перевода задачи в статус «Выполнена» **infrastructure.TaskStates** формируется запись для задачи со статусом **Completed.**
9. Кнопка «**Отмена**» выполнения задачи для перевода в статус **Cancelled**.
10. Кнопка «**Из шаблона**» работ по классу оборудования по типу обслуживания. Заполнить перечень задач и их параметры на основе шаблона работ, привязанного к типу обслуживания класса оборудования этой единицы оборудования.

#### Компонент «Отчет по работе»

Jira:

Является дочерним компонентом «**Задачи по оборудованию**». Предназначен для формирования отчета по выполненной работе в виде фотографии, заметки, чек-листа, а также подробного учета времени по исполнителям.

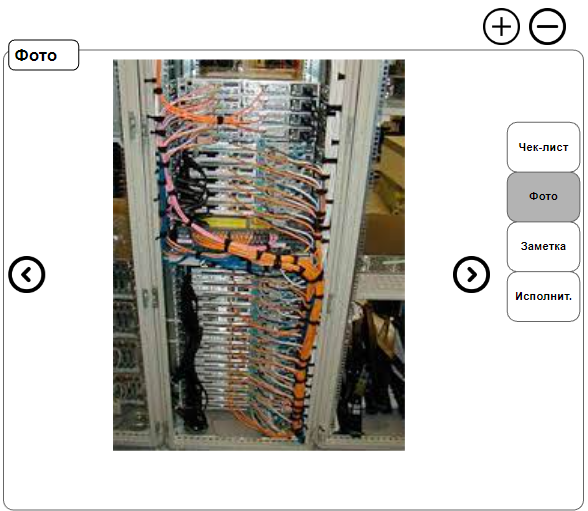


Пользователь может переключаться между экранами компонентов:

1. Чек-список.
2. Фото.
3. Заметка.
4. Исполнители.

#### Компонент «Задача. Фото»

Jira:



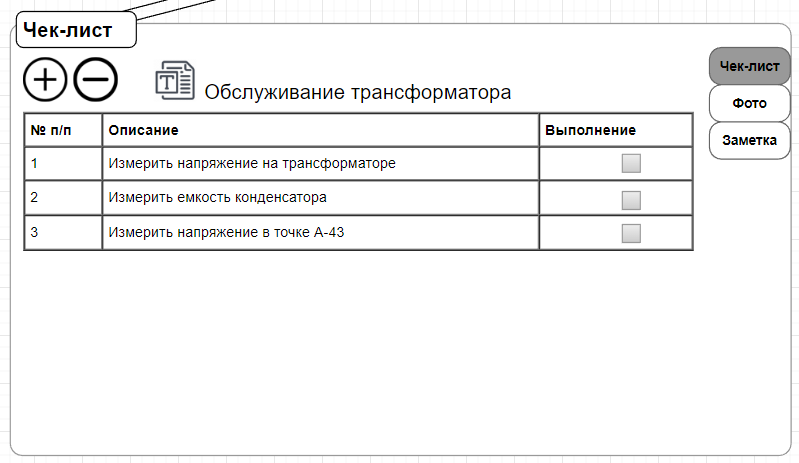
Компонент предназначен для регистрации состояния оборудования в виде набора фотографий.

Действия:

1. Добавить фотографию. На мобильном устройстве вызывается приложение для выполнения фотографирования.
2. Удалить фотографии.
3. Показать предыдущую фотографию.
4. Показать следующую фотографию.

#### Компонент «Задача. Чек-лист»

Jira:

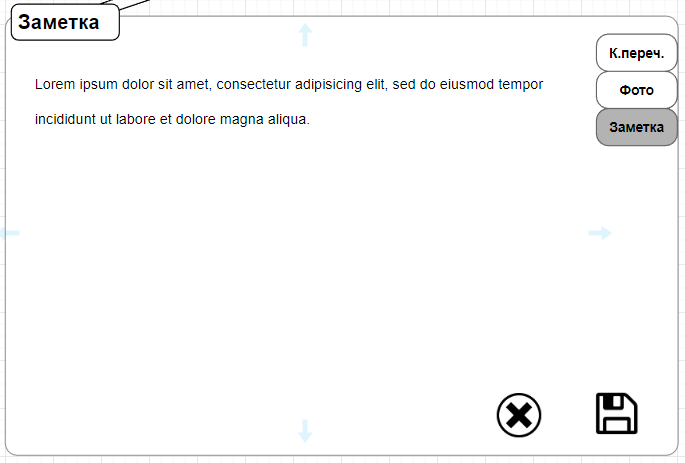


Компонент предназначен для выполнения задач по чек-листу.

Уточнить, зачем он вообще нужен.

#### Компонент «Задача. Заметка»

Jira:



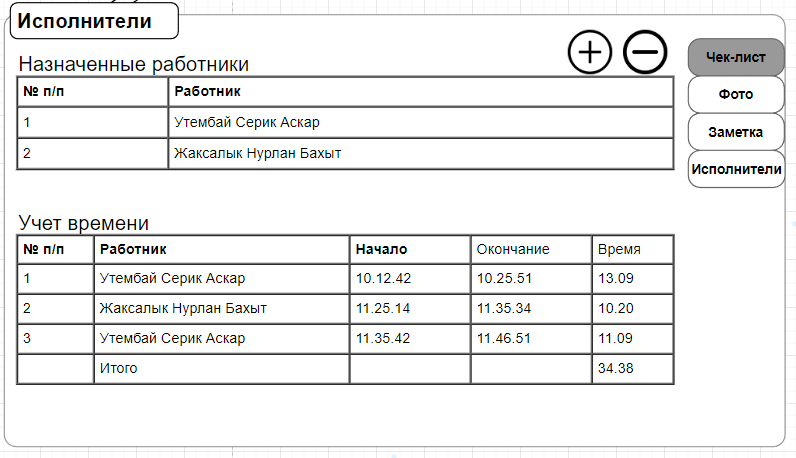
Компонент предназначен для предоставления возможности оставить заметку о выполнении задачи в текстовой форме.

Действия:

* 1. Набрать текст заметки. Простой редактор.
  2. Сохранить заметку.
  3. Удалить заметку.

#### Компонент «Задача. Исполнители»

Jira:



Компонент предназначен для назначения работников на задачу и учет их времени работы.

Состоит из двух табличных частей.

Табличная часть «**Назначенные работники**»:

Действия:

1. Добавить работника. При заполнении назначенных работников выбор проводится по списку работников, которые закреплены за наряд-заданием (см. компонент «**Работники**»).
2. Удалить пользователя.
3. Если список работников наряд-задания пустой, то работу выполняет только «**Ответственный**».

Нужен ли допуск по списку назначенных работников? Вариант: если назначенные есть, то только они могут выполнять работу. Если назначенных нет, то любой, кто входит в число работников по наряд-заданию, может выполнять работу.

Табличная часть «**Учет времени**».

Выводится информация по учету времени при выполнении задачи. Учет времени заполняется на основе изменения статуса задачи. Старт задачи — начало, останов — окончание периода работы.

Действия.

==================================================================

**Review˅**

* 1. **Маршрут по оборудованию - EqRoute**. Если включен – то в секции выполнения работ появляется список оборудования, в котором можно задать маршрут из оборудования, который должен быть обслужен по этому документу. При этом поле «Единица оборудования» на закладке «Основные данные» становится опциональным. Если оно было заполнено, то его значение переносится в список маршрута, а само поле очищается.
  2. **Разные классы оборудования - DiffEqClasses**. Опция активна, если выбрана опция «Маршрут по оборудованию». Если выбрана эта опция, то поле «**Класс оборудования**» на закладке «**Основные данные**» очищается и в маршруты можно добавлять оборудование разных классов оборудования. Если опция не выбрана, то в маршрут можно добавлять только оборудование с классом, указанным в «**Класс оборудования**».
  3. **Смены - MultiShifts**. Если опция включена, то становится доступной закладка «Смены», где можно настроить работу по сменам. ToDo.
  4. **Инспекции - EqInspection**. Если опция включена, то при регистрации работ по оборудованию становится доступной закладка «**Инспекции**», где можно зарегистрировать показатели измерений характеристик оборудования.
  5. **Периодическое наряд**-**задание — PeriodicalDoc**. Если опция включена, то становится доступной закладка «**Периодичность**», где можно настроить период повторения данной работы.
  6. **Допуск по технике безопасности (ТБ) - SafetyAccess**. Если опция включена, то становится закладка «**Допуск по ТБ**». На закладке можно зафиксировать мероприятия по допущению работников к выполнению работ.

##### Связанные документы.

Jira:

<http://jira.abitech.kz:8080/browse/AS-42>

Сюда записываются ссылки на документ, на основе которого был создан текущий документ, а также все документы, которые были созданы на основе данного.

**Шаблон работы** – ссылка на шаблон, на основе которого создан текущий документ, если он создан на основе шаблона.

Табличная часть:

* 1. **№ п/п** – номер по порядку.
  2. **Вид** – «**Родитель**» / «**Дочерний**».
  3. **Тип** – «**Тип документа**». Ссылка на типы документов модуля «**Рабочий процесс**».
  4. **Документ** – ссылка на документ. **ToDo**: продумать, чтобы переходе на документ открывалась форма для нужного типа документа. Возможно, в справочнике типов документов хранить имя компоненты, которая отвечает за документ.

Данные хранить в таблице **DocReferences** модуля **DocWorkflow**. Структура таблицы:

* + 1. **Id** – Guid – записи могут создаваться в режиме офлайн.
    2. **Type** – string –
       1. Parent – родитель.
       2. Child – дочерний.
    3. **DocTypeId** – тип документа – ссылка на справочник **DocTypes** модуля **DocWorkflow**.
    4. **DocRef** – Guid – ссылка на документ.

##### Подписи.

Jira:

<http://jira.abitech.kz:8080/browse/AS-43>

Здесь собираются подписи или согласовано от работников, которые отвечают за проведения работы. Для этого используется табличная часть:

* 1. **№ п/п** – номер по порядку.
  2. **Роль по документу**.
  3. **Должность** – ссылка на справочник должностей.
  4. **Работник** – ссылка на справочник работников.
  5. **Согласовано** – кнопка. Кнопка изначально находится в состоянии «**Нет**». Текущий пользователь может перевести кнопку в состояние «**Да**» напротив себя как работника. После этого запись для работника становится закрытой от редактирования. При необходимости поменять работника в статусе «**Согласовано**» необходимо удалить его запись, добавить правильную, работник должен согласовать.

Данные хранить в таблице **DocSigns** модуля **DocWorkflow**. Структура таблицы:

* + 1. **Id** – Guid – записи могут создаваться в режиме онлайн.
    2. **DocRole** – string – роль по документу. Заполняется вручную.
    3. **UserPositionId** – должность пользователя. Ссылка на справочник **hr.Positions.** Заполняется на основе поля «**Работник**». Из справочника **hr.Positions** предоставить для выбора записи, которые относятся к работнику (через линк-таблицу **UserPositions**. Проверить, что есть/нет функция, которая возвращает позиции по работнику).
    4. **UserId** – работник. Ссылка на справочник **hr.Users**.
    5. **IsAccepted** – булево значение. Признак, что пользователь согласовал документ. По умолчанию – **False**. Пользователь может проставить значение **True** только для записей таблицы, которые относятся к этому пользователю.

##### История маршрута документа.

Jira:

Здесь отображается история изменений статуса документа от создания до закрытия. Для этого используются данные таблицы **docworkflow.DocExecLog**.

* 1. **№ п/п** – номер по порядку.
  2. **StatusStartTime - Время изменения** – время изменения статуса документа.
  3. **StatusStartUserId - Исполнитель** – ссылка на справочник работников. Кто изменил статус документа.
  4. **­StatusId - Статус** – статус, на который перевели документ. Ссылка на справочник статусов документа модуля «**Рабочий процесс**».

1. **Смены**.

Здесь настраиваются смены, которые участвуют в выполнении работы.

##### Периодичность.

Здесь задается периодичность повторения данной работы.

1. Допуск по технике безопасности.

Здесь фиксируются процедуры по допуску работников к выполнению работы.

#### Секция регистрации информации по выполнению работы – операциям и оборудованию

Секция содержит набор элементов для планирования операций работы и фиксации хода ее выполнения.

##### Табличная часть «Маршрут оборудования»

Jira:

http://jira.abitech.kz:8080/browse/AS-61

Маршрут по оборудованию определяет перечень оборудования к обслуживанию и последовательность обслуживания оборудования. К позициям маршрута прикреплена форма детализации хода обслуживания единицы оборудования.

Табличная часть состоит из колонок:

1. № п/п.
2. **Id** – не отображается.
3. **JobId** – ссылка на текущий документ наряд-задания.
4. **AssetId** – оборудование — ссылка на справочник оборудования **inventory.Assets**.
5. **MaintenanceTypeId** – вид обслуживания — вид обслуживания оборудования, если есть. Выбирается из справочника **infrastructure**.**MaintenanceTypes**. Если для класса оборудования есть записи, то предлагается список с фильтром по этому классу. Пользователь может очистить фильтр, чтобы получить доступ к полному списку типов обслуживания.
6. **Класс** — класс оборудования — ссылка на справочник **inventory.AssetClasses**. Выводится в качестве справки в таблице маршрута оборудования не сохраняется.

Маршрут оборудования сохраняется в таблице infrastructure.**JobRoutes**.

Предусмотреть возможность добавления и удаления позиций оборудования в маршрут оборудования.

Предусмотреть возможность изменения порядка единицы оборудования в списке (перемещение вверх/вниз по списку).

Предусмотреть кнопку копирования настроек работ по единице оборудования на остальные единицы оборудования. Активно только в статусе документа **Draft**.

##### Табличная часть «Задачи»

Jira:

Предназначена для формирования списка задач/работ по наряд-заданию и отслеживания статуса их выполнения.

------------------------------------------------------------------------------------------------

Детализация: «**Чек-лист**».

Jira:

Чек-лист представляет собой табличную часть со следующими колонками:

1. № п/п.
2. Описание — **Description**.
3. Отметка о выполнении — **IsDone**.

На панели инструментов нужны кнопки для

1. Добавления позиции чек-листа.
2. Удаления позиции.
3. Заполнения табличной части из шаблона чек-листа.

Таблицы для хранения информации по работе с чек-листами:

1. Шаблоны чек-листов — таблица **infrastructure.CheckLists**.
2. Позиции шаблонов — таблица **infrastructure.CheckListItems**.
3. Результат заполнения чек-листов — таблица **infrastructure.CheckListResults**. Структура таблицы:
   1. **Description** – строка — описание. Добавить поле.
   2. **CheckListItemId** – ссылка на позицию шаблона чек-листа **CheckListItems**.
   3. **TaskId** – ссылка на задачу работы **Tasks**.
   4. **AssetId** – ссылка на единицу оборудования **inventory.Assets**.
   5. **EmployeeId** – пользователь, который проставил значение в поле **IsDone**. Cсылка на таблицу **hr.Users**.
   6. **IsDone** – булево поле — признак, что операция выполнена.

------------------------------------------------------------------------------------------------

Детализация «**Фото**»

Jira:

Фото представляет собой галерею фотоснимков, которые можно пролистать, добавить, удалить. При добавлении можно сделать фото или выбрать с устройства хранения.

Ссылки на фото хранятся в таблице **infrastructure.Attachments**;

------------------------------------------------------------------------------------------------

Детализация: «**Исполнители**»

Jira:

Детализация представляет собой две табличные части: назначенные работники и учет времени по исполнителям. Записи в табличные части можно добавлять вручную. Хотя для учета времени — это скорее исключение.

------------------------------------------------------------------------------------------------

Детализация: «**Заметка**»

Jira:

Простой редактор текста, где исполнитель может оставить любое текстовое примечание к выполнению работы.

Текст хранится в таблице **infrastructure.EquipmentNotes**. ------------------------------------------------------------------------------------------------

------------------------------------------------------------------------------------------------

1. Замечания.
2. Отчеты — отмечаются, какие типы отчетов должны быть выполнены:
   1. Чек-лист.
   2. Фото.
   3. Заметка.
   4. Показатели по оборудованию.

##### Табличная часть «Материалы»

Табличная часть предназначена для формирования перечня необходимых материалов по номенклатуре ТМЦ предприятия с указанием потребности и фактически использованных.

В табличной части можно добавить и удалить записи по материалу.

Количество «**Норма**» заполняется на этапе редактирования документа в статусе «**Черновик**». После изменения статуса это поле становится только для чтения.

Данные по табличной части хранятся в таблице WOrderEqMaterials, структура которой:

* 1. **Id** – Guid – может заполняться на мобильном устройстве.
  2. **WOrderId** – ссылка на этот документ – infrastructure.Jobs.
  3. **EqId** – ссылка на единицу оборудования – inventory.Assets.
  4. **NomenclatureId** – позиция номенклатуры – ссылка на inventory.Nomenclature.
  5. **UoMId** – единица измерения – ссылка на таблицу hr.UnitOfMeasures.
  6. **StdQnt** – double – количество по норме.
  7. **­­­ActualQnt** – double – фактическое количество.

##### Табличная часть «Измерения»

Табличная часть предназначена для регистрации результатов измерения параметров, характеристик единицы оборудования. На экране также приведены нормальные характеристики для справки.

Строки таблицы могут быть добавлены, удалены или заполнены на основании шаблона измерений для класса оборудования и типа обслуживания.

Нормативные показатели находятся на раскрывающейся/закрывающейся панели с кнопками для раскрытия/закрытия этой панели.

Колонки табличной части:

1. **№ п/п** – номер строки в таблице.
2. **Id** – Guid – записи могут создаваться в режиме офлайн. Не отображается
3. **TaskId** – задача - ссылка на эту задачу. Не отображается.
4. **ReadingsId** – показатель – выбирается из таблицы **infrastructure.ReadingTypes**.
5. **UoMId** – единица измерения – выбирается из таблицы единиц измерения – **hr.UnitsOfMeasure**.
6. **ReadingsValue** – строка – показатель.
7. **ReadingsStandardId** – нормативные показатели для единицы оборудования – ссылка на таблицу **EqReadingsStandarts.** На экране отображать колонки для
   1. **Отношение** – **ComparisonType.**
   2. **Норма мин. – MinValue.**
   3. **Норма макс.** – **MaxValue**.

Результаты измерений по табличной части хранятся в таблице **infrastructure.WOrderEqReadings**.

##### Табличная часть «Руководства»

Табличная часть предназначена для добавления руководств, инструкций, чертежей и т.д., необходимых для выполнения задачи.

В табличной части можно добавить запись для руководства, удалить запись, скачать документ на рабочее устройство для дальнейшего просмотра.

Колонки табличной части:

1. **№ п/п** – порядковый номер записи.
2. **Id** – Guid – записи могут создаваться в режиме офлайн. Не отображается.
3. **TaskId** – задача – ссылка на текущую задачу наряд-задания. Не отображается.

#### Табличная часть «Бригада»

Предназначена для ведения учета работников, которые выполняют работы по этому наряд-заданию.

Колонки табличной части:

1. № п/п.
2. ФИО.
3. Должность.
4. Квалификация.
5. Допуск к работе — заполняется .

Состав бригады заполняется при заполнении поля «Бригада» в шапке документа. При этом начальник бригады заполняется в «Производящие работы», остальные члены бригады — в состав бригады.

Должна быть возможность добавления членов бригады из справочника «Работники».

ФИО, квалификация, должность заполняются данными из справочника.

#### Табличная часть «Материалы»

Предназначена для нормирования и учета движения ТМЦ при выполнении работ по наряд-заданию.

Колонки табличной части:

1. № п/п.
2. Номенклатура — выбирается из справочника «Номенклатуры ТМЦ».
3. Ед. измерения — заполняется из данных позиции номенклатуры.
4. Количество «норма» — заполняется планировщиком.
5. Количество «использовано» - заполняется ???

#### Табличная часть «Оборудование»

Предназначена для планирования необходимости использования и периода занятости оборудования, в том числе специального.

Колонки табличной части:

1. № п/п.
2. Оборудование — выбирается из справочника «Оборудование / ОС».
3. Начало работ — плановое начало использования оборудования.
4. Окончание работ — плановое окончание использования оборудования.

### Стандартные статусы «Наряд-задания»

#### Статус «Черновик»

При создании документа ему присваивается статус «Черновик». В этом статусе документ можно редактировать.

### Доступ к функциональности документа

Доступ к функциональности документа задается с учетом функциональности по доступам модуля «Рабочий процесс».

## Повторяющееся простое наряд-задание (legacy)

## Журналы

### Журнал запросов на обслуживание

Журнал предназначен для регистрации запросов на проведение обслуживания оборудования, в том числе созданных системой.

Модуль: **ТОиР**

Название таблицы журнала: **MaintRequestRegistry**.

Меню: ТОиР > Журналы > Журнал запросов на обслуживание.

Структура справочника:

* 1. **Id** – Guid – записи могут создаваться на мобильных устройствах в офлайн режиме.
  2. **InitDateTime** – дата и время формирования запроса на обслуживание.
  3. **EqId** – ссылка на единицу оборудования. ОС – Активы/оборудование.
  4. **MaintenanceTypeId** – ссылка на справочник типов обслуживания.
  5. **ReadingsTypeId** – тип показателей. Ссылка на справочник типов показателей оборудования.
  6. **InitReadingsValue** – значение показателя на момент формирования запроса.
  7. **InitReadingsUoMId** – единица измерения показателя на момент формирования запроса.
  8. **InitWorkerId** – инициатор запроса, ссылка на справочник «**Работники**».
  9. **MaintStartDateTime** – дата и время начала обслуживания оборудования. Заполняется в момент, когда наряд-задание переводится в статус «**Начаты работы**».
  10. **MaintStartReadingsValue** – значение показателя на момент начала обслуживания.
  11. **MaintStartReadingsUoMId** – значение единицы показателя на момент начала обслуживания.
  12. **MaintDocRef** – ссылка на документ, на основе которого проводится обслуживание.
  13. **MaintDoneDateTime** – дата и время завершения обслуживания оборудования. Заполняется в момент, когда наряд-задание переводится в статус «**Закрыто**».
  14. **RequestStatusId** – текущий статус заявки на обслуживание. Обновляется по мере выполнения работ над запросом и наряд-заданием. Ссылка на **DocStatuses** модуля **Workflow**.

Требования к экранной форме: обычная функциональность, плюс:

* + 1. Возможность установки фильтра по полю.
    2. Возможность выборки за период по полям:
       1. **InitDateTime.**
       2. **MaintStartDateTime**.
    3. При заполнении карточки записи вручную ссылка на документ-родитель не требуется (**MaintDocRef**).

Будущее развитие:

1. Планируется добавление документа «Заявка на обслуживание». Появятся поля-ссылки на документ.

### Журнал результатов измерения параметров/показателей оборудования. (инспекции) оборудования

Журнал предназначен для фиксации результатов осмотра / инспекции единиц оборудования их узлов.

Модуль: **ТОиР**.

Название таблицы БД: **EquipInspections**.

Меню:

* En: Maintenance > Registries > Equipment Inspections.
* Ru: ТОиР > Журналы > Результаты осмотров.

Структура таблицы:

* 1. **Id** – Guid – записи могут генерироваться на мобильном клиенте офлайн.
  2. **EqId** – единица оборудования. Ссылка на ОС/Активы/оборудование (модуль инвентаризация). Копируется из документа инспекции.
  3. **InspectionDateTime** – дата и время проведения обследования. Копируется из документа инспекции.
  4. **InspectionDocRef** – документ инспекции. Ссылка на документ инспекции/осмотра оборудования.
  5. **ReadingsTypeId** – тип показателя. Ссылка на справочник типов показателей. Копируется из документа инспекции.
  6. **ReadingsValue** – строка - значение показателя. Копируется из документа инспекции.
  7. **ReadingsUoMId** – единица измерения. Ссылка на справочник видов единиц измерения. Копируется из документа инспекции.

Пользовательский интерфейс: стандартная таблица данных. Возможность задавать период по полям типа Date+Time.

В развитие:

* 1. Добавить возможность заполнения показателей на основе измерительных приборов.
  2. Добавить возможность прикрепления фотоснимков к показанию.

### Журнал проведенных работ по обслуживанию оборудования (AS-2)

Jira:

<http://jira.abitech.kz:8080/browse/AS-2>

Журнал предназначен для фиксации проведенных работ по единице оборудования. Заполняется на момент завершения работы по обслуживанию.

Название таблицы:

**EqMaintJournal**

Структура журнала:

1. **Id** – Guid – запись может быть сформирована на основе закрытия наряд-задания на мобильном клиенте.
2. **ReqRegId** – ссылка на запись журнала «**Запрос на обслуживание**», если обслуживание инициировано запросом на обслуживание.
3. **EqId** – единица оборудования. Ссылка на справочник «**Активы/оборудование**».
4. **MaintDoneDateTime** – дата и время выполнения запроса.
5. **MaintDocType** – документ, на основе которого закрыт запрос. Тип документа. **DocTypes** модуля **Workflow**.
6. **MaintDocId** – документ, на основе которого закрыт запрос. Ключ записи.
7. **MaintResponsibleId** – ответственный за проведение работ. Ссылка на справочник «**Работники**».
8. **MaintTypeId** – тип обслуживания. Ссылка на справочник «**Типы обслуживания**».
9. **ReadingsTypeId** – тип показателей. Справочник на типы показателей оборудования.
10. **ReadingsValue** – значение показателя на момент начала / окончания обслуживания.
11. **ReadingsUoMId** – единица измерения показателя. Ссылка на справочник «**Единицы измерения**».
12. **MaintPlanStartDateTime** – дата и время начала обслуживания.
13. **MaintStartDateTime** – дата и время начала обслуживания.
14. **MaintPlanDoneDateTime** – дата и время начала обслуживания.

Функциональность на пользовательском интерфейсе – отображение списка записей, без формы работы с записью. Объем функциональности – стандартный, плюс:

* 1. Возможность выборки за период по полям:
     1. **MaintDoneDateTime.**

### Журнал наряд-заданий

Предназначен для регистрации наряд-заданий и операций над ними.

### Журнал показаний работы оборудования (EntityReadings)

Журнал предназначен для регистрации физических показателей, которые описывают работу оборудования, или показывают его состояние.

Структура журнала:

* 1. Id – системный ключ записи.
  2. ReadingTypeId – тип показателя.
  3. EntityType – тип сущности
  4. EntityGuidId – ключ записи сущности
  5. EntityIntId – ключ записи сущности (целочисленный)
  6. Value – значение показателя.
  7. OccuredAt – время снятия показаний.
  8. Reference – ссылка на источник данных. Например, при определении пробега — на запись в журнале трекинга объектов.
  9. Notes – примечания.
  10. AssetClassId – класс оборудования/основных средств – ссылка на справочник классов оборудования (модуль inventory).
  11. IsNormalValue – bool – признак, что значение является в пределах нормы. Если не в пределах нормы, то заполняется журнал.

### Журнал регистрации дефектов оборудования (AS-109)

Jira:

<http://jira.abitech.kz:8080/browse/AS-109>

Журнал предназначен для регистрации дефектов оборудования, обнаруженных при осмотрах или при измерениях параметров оборудования.

Таблица: **EqDefectsJournal** – модуль infrastructure.

Структура журнала:

1. **Id** – Guid – записи могут формироваться в режиме офлайн.
2. **AssetId** – ссылка на справочник основных средств/оборудования, по которому обнаружен дефект.
3. **AssetClassId** – ссылка на класс оборудования. Заполняется из записи единицы оборудования (**Assets**).
4. **DocId** – Guid – ссылка на документ – источник информации по дефекту. Может быть пустым.
5. **DocTypeId** – int – ссылка на тип документа, модуль **DocWorkflow**. Пусто, если **DocId** не заполняется.
6. **DefectTypeId** – int – тип дефекта – ссылка на справочник дефектов. Может быть пустым, если тип дефекта не установлен.
7. **DefectNote** – string – описание дефекта, если дефект не стандартизован по типам дефектов.
8. **DefectQnt** – int – количество дефектов. Например: «для излома триангеля – **2** шт».
9. **DefectResolutionNote** – string – описание предпринятых мер.
10. **MaintRequestId** – Guid – ссылка на запись в журнале заявок на обслуживание. Заполняется, когда на основе решения по дефекту формируется заявка на обслуживание оборудования.
11. **OpenDT** – дата и время обнаружения/фиксации дефекта.
12. **ResolveDT** – дата и время устранения дефекта.
13. **IssueAuthorId** – кто обнаружил/зафиксировал дефект. Ссылка на **Users** модуля **HR**.

Пока реализация в простом варианте. Пользователь может

* 1. Добавить, отредактировать, удалить запись.
  2. Отсортировать по полю.
  3. Выбор позиций из связанных справочников – через простой ниспадающий список.

### Журнал регистрации результатов осмотров (AS-110)

Jira:

http://jira.abitech.kz:8080/browse/AS-110

Журнал предназначен для регистрации результатов осмотров оборудования. Если обнаружены отклонения от нормы в виде дефектов, то они регистрируются также в журнале дефектов.

Название таблицы: **EqInspectionResults** – модуль **infrastructure**.

Структура:

1. **Id** – Guid – запись может создаваться в режиме онлайн.
2. **AssetClassId** – ссылка на класс оборудования/основных средств. Модуль Inventory. Заполняется из записи единицы оборудования (**Assets**).
3. **AssetId** – ссылка на единицу оборудования. Assets модуля inventory.
4. **InspectionNote** – string – описание результата осмотра. Может быть пустым.
5. **DefectNote** – string – описание дефекта, если есть. Может быть пустым.
6. **DefectTypeId** – ссылка на тип дефекта – справочник типов дефектов EqDefectTypes. Может быть пустым.
7. **DefectQnt** – int – количество дефектов.
8. **EqDefectsJournalId** – ссылка на запись журнала дефектов.
9. **ResponsibleId** – ответственный за осмотр – ссылка на Users модуля HR. По умолчанию - текущий пользователь системы.
10. **InspectionDT** – время регистрации осмотра. По умолчанию – текущее время.
11. **DocId** – ссылка на запись документ, если запись формируется на основе документа.
12. **DocTypeId** – тип документа инспекции – ссылка на справочник типов документов модуля **DocWorkflow**.

Функциональность:

1. CRUD.
2. Сортировка по полям.
3. Фильтры по полям.
4. При создании записи, если заполнены поля, которые относятся к описанию дефекта, то выполняется регистрация записи в журнале дефектов **EqDefectsJournal**. Ключ записи журнала дефектов прописывается в поле **EqDefectsJournalId.** Соответствие полей:
   1. EqDefectsJournal.AssetId = EqInspectionResults.AssetId.
   2. AssetClassId = AssetClassId.
   3. DocId = DocId.
   4. DocTypeId = DocTypeId.
   5. DefectTypeId = DefectTypeId.
   6. DefectNote = DefectNote.
   7. DefectQnt = DefectQnt.
   8. UssueAuthorId = ResponsibleId.
5. Выбор ссылок на другие справочники – из простых ниспадающих списков.

### Журнал регистрации отклонений показателей от нормы.

## Бизнес-процесс формирования заявки на обслуживание на основе периода

## Бизнес-процесс формирования заявки на обслуживание на основе показателей оборудования

Бизнес-процесс предназначен для регистрации показателей оборудования для дальнейшего анализа работы оборудования с точки зрения формирования работ по его обслуживанию, эффективности использования, а также производственного эффекта. Регистрируются показатели, которые характеризуют физическое окружение оборудования (влажность среды, химическая экспозиция, …), выработка оборудования (пробег, моточасы, количество выпущенной продукции, …), показания приборов, которые характеризуют состояние оборудования (вольтаж, ампераж, …). Не регистрируются календарные периоды (масло масляное).

Входы процесса:

1. Показатели оборудования. В том числе полученные на основе других приборов/оборудования. Показатели могут быть собраны как работниками, так и автоматически. Показатели оборудования регистрируются в **Журнале показателей оборудования**.

Шаги процесса:

1. Снятые показатели регистрируются в **Журнале показателей оборудования**.
2. На основе правил формирования работ по обслуживанию формируется **Заявка на обслуживание** в **Журнале заявок на обслуживание оборудования**. Если правила не отработали, то процесс заканчивается. Пример правил:
   1. Пробег превысил 10000 км с момента последнего обслуживания.
   2. Напряжение должно соответствовать 220 В ± 10 В.
3. Планировщик просматривает журнал и формирует
   1. Дефектную ведомость.
   2. Или наряд-задание на работы по обслуживанию.
4. Производящий работы/ответственный выполняет работы.
5. Когда дефектная ведомость или наряд-задание закрывается, то в **Журнал заявок на обслуживание оборудования** вносится информация по выполнению работ по обслуживанию.

Выходы процесса:

1. Журнал заявок на обслуживание оборудования.

Роли процесса:

1. Механик оборудования.
2. Планировщик.
3. Производящий работы / бригадир / ответственный.

### Заявка на обслуживание оборудования

Заявка предназначена для формирования списка оборудования, по которому необходимо провести обслуживание (по решению составителя заявки) с указанием причины, выявленных проблем и рекомендуемых операций (по решению планировщика работ).

Модуль: ТОиР

Название таблицы в базе данных: **SerReqs**.

Пункт меню: "ТОиР" > "Заявка на обслуживание".

Документ заявки состоит из следующих частей.

Шапка документа:

1. **Id** заявки — GUID.
2. **DocDateTime** – Дата+Время создания документа (дата + время операции формирования заявки на обслуживание).
3. **Description** – Описание заявки.
4. **WorkFlowTypeId** – тип маршрута документа, по которому должна двигаться заявка.
5. **WorkFlowStatusId** – текущий статус документа по отношению к маршруту документа.

#### Табличная часть: Оборудование/ компонента оборудования

Табличная (+ древовидная) часть предназначена для формирования списка единиц оборудования, по которым необходимо провести обслуживание.

Название таблицы в базе данных: **ServReqEqItems**.

Структура табличной части имеет древовидную форму и имеет следующие колонки:

1. № п/п — (вычисляется) текущий порядковый номер строки в списке.
2. **EquipmentId** – ссылка на таблицу **Основные средства** / **Оборудование** (Inventory>Assets).
3. Описание оборудования — поле **Description**.
4. Класс оборудования — (вычисляется) — ссылка на справочник классов оборудования (AssetClass).
5. **MaintenanceTypeId** – ссылка на справочник типов обслуживания MaintenanceTypes.
6. **ReasonNote** – примечание — описание причины требуемого обслуживания.
7. ParentId – ссылка на родительскую запись

#### Табличная часть "Операции по обслуживанию"

Табличная часть предназначена для формирования списка операций, необходимых для обслуживания единицы оборудования или узла. Таким образом табличная часть соотносится с записью табличной части "Оборудование / компонента оборудования".

Название таблицы в базе данных: **ServReqEqItemOpers**.

Структура:

1. №п/п — порядковый номер.
2. **Description** – описание операции.
3. **Responsible** – исполнитель. Ссылка на справочник работников предприятия.
4. **StdOperId** – ссылка на справочник "Стандартные операции". Будущее развитие.

#### Табличная часть "Материалы на обслуживание"

Табличная часть предназначена для формирования списка операций, необходимых для обслуживания единицы оборудования или узла. Таким образом табличная часть соотносится с записью табличной части "Оборудование / компонента оборудования".

Название таблицы в базе данных: **ServReqEqItemInvents**.

Структура:

1. №п/п — порядковый номер.
2. **Description** – строка описание материала.
3. **Quantity** – число - количество.

Формуляр **Журнала запросов на обслуживание**:

1. Единица оборудования.
2. Дата и время запроса на обслуживание. Для автоматической регистрации на основании показаний датчиков или времени - время генерации/активации наряд-задания.
3. Дата и время начала обслуживания - когда наряд-задание переведено в статус “В работе”.
4. Дата и время проведения обслуживания - когда наряд-задание было переведено в состояние “Закрыто”.
5. Вид обслуживания.
6. Класс оборудования.
7. Тип показателя - тип показателя, на основе которого было принято решение о проведении обслуживания (давление, температура, пробег).
8. Ед. измерения. Нужно только для экранной формы в качестве информации.
9. Величина показателя на момент запроса. Заполняется на момент формирования записи в Журнале.
10. Величина показателя на момент начала обслуживания. Регистрируется значение показателя, когда наряд-задание переводится в статус “В работе”.
11. Запрос на обслуживание - ссылка на документ Запрос на обслуживание, если запрос создан на основе этого документа.
12. **Автор запроса** — ссылка на справочник работников.
13. Рабочее задание - ссылка на наряд-задание на проведение обслуживания, которое создается для выполнения запроса на обслуживание.
14. Статус - текущий статус заявки на обслуживание.
    1. “Ожидает обслуживания”.
    2. “Обслуживается”.
    3. “Обслужено”.

Порядок заполнения Журнала:

1. Журнал может заполняться вручную или системой на основе показаний датчиков.
2. Заполнение производится в два этапа:
   1. При возникновении условий для проведения оборудования – информация для формирования/активации наряд-задания.
   2. При закрытии наряд-задания на проведение обслуживания, зарегистрированного в этом журнале – для отчета по выполнению Заявки.

## План работы на период

План работы на период — это совокупность всех запланированных наряд-заданий на этот период. Сюда включаются:

1. Обычные наряд-задания, которые запланированы на будущую дату.
2. Повторяющиеся наряд-задания, которые повторяются через определенные периоды времени.
3. Повторяющиеся наряд-задания, которые повторяются на основе таких показателей оборудования, как пробег, отработанные мото-часы и прочее. В этом случае используются средние показатели оборудования.

### Генерация плановых наряд-заданий на период

Генерация повторяющихся наряд-заданий выполняется из Журнала запланированных наряд-заданий. Формирование выполняется на период, который указывает пользователь. По умолчанию на год от текущей даты.

Системой формируется перечень оборудования, для которого выполняются работы на основе «Видов обслуживания», у которых есть условия обслуживания по:

1. Периоду обслуживания.
2. Пробегу.
3. Выработке.
4. Других показателей, кроме условий обслуживания, которые относятся к обслуживанию по состоянию или на основе предсказуемых моделей обслуживания.

Если для вида обслуживания существует два и больше условия для проведения обслуживания, то используется тот, у которого стоит признак: «использовать при планировании». Если такого признака нет, то используется первое произвольное.

Система последовательно перебирает единицы оборудования, виды обслуживания и для каждого вида обслуживания формирует наряд-задание согласно условиям проведения обслуживания.

Для периодических работ формируются наряд-задания в пределах указанного периода планирования (например, год).

# Internet of Things (IoT) / SCADA

Модуль предназначен для сбора данных с датчиков, управления исполняющими устройствами, контроль работы датчиков и исполняющих устройств.

Работу модуля IoT рекомендуется выполнять по Event-driven architecture.

## Общая схема

[IoT\_Module.drawio - diagrams.net](https://app.diagrams.net/" \l "Htvolodi%2Fabitech%2Fmain%2FIoT%2FIoT_Module.drawio)

## Справочник контроллеров

## Логирование показаний датчиков

# Модуль "Трекинг объектов" (tracking) / Логистика / Транспорт

## Справочники

### Справочник "Маршруты"

Jira:

<http://jira.abitech.kz:8080/browse/AS-4>

Справочник маршруты предназначен для регистрации настроек маршрутов транспорта/движущихся объектов в системе.

Модуль: **Трекинг**.

Меню: **Справочники**.

Название справочника в базе данных: **Routes**.

Структура справочника:

1. Id
2. Name – название маршрута.

### Справочник типов транспортных средств

Jira:

<http://jira.abitech.kz:8080/browse/AS-7>

Справочник типов транспорта предназначен для ведения типов транспортных средств.

Модуль: Трекинг.

Меню: Справочники.

Название в базе данных VehicleTypes.

Структура справочника:

1. Id – целое значение.
2. Description – описание типа.
3. AssetClassId – ссылка на справочник Класс оборудования (основных средств).

Функциональность справочника — стандартная.

### Справочник контейнеров

Структура справочника является иерархической:

1. Id
2. Code – строка. Уникальное значение.
3. Description – описание. Обязательно к заполнению.
4. ParentId – ссылка на родительскую запись.

Интерфейс справочника:

1. Отображается совместно с позициями справочника «**Транспортные средства**».
2. Должна быть возможность добавлять новые позиции справочника как дочерние элементы.
3. Должна быть возможность удалять элементы дерева, при этом дочерние элементы также удаляются.
4. Должна быть возможность изменять название элементов.
5. Должна быть возможность «перетаскивания» элементов с одной ветки дерева на другую.
6. Справочник является фильтром для отображения элементов справочника «**Транспортные средства**».

### Справочник «Транспортные средства»

Справочник предназначен для учета единиц транспортных средств (вагонов, авто, самолетов, кораблей, вьючных верблюдов, …) в разрезе типов транспортных средств.

Модуль: **Трекинг**.

Название таблицы: **Vehicles**.

Меню: «**Трекинг**» > «**Справочники**» > «**Транспортные средства**»

Структура справочника:

1. **Id**
2. **PlateNumber** – строка - Регистрационный номер.
3. **Description** – описание.
4. **VehicleTypeId** – ссылка на справочник типов транспортных средств.

Интерфейс справочника:

1. Позиции справочника отображаются совместно с элементами справочника «**Типы транспортных средств**». При навигации по «**Типам** …» элементы справочника «**Транспортные средства**» соответственно отфильтровываются по выбранному типу.
2. Должна быть возможность поиска, сортировки и т.д. как у обычного справочника.

### Справочник «Транспортные составы»

Справочник предназначен для ведения составных транспортных средств: ж/д составов, автомобильных составов.

Модуль: **Трекинг**.

Название таблицы:

1. Шапка справочника: **ComposedVehicles.**
2. Позиции справочника**: ComposedVehicleItems.**

Меню: «**Трекинг**» > «**Справочники**» > «**Транспортные составы**».

Структура справочника.

Справочник состоит из шапки записи состава и табличной части, которая отображает транспортные единицы состава.

Шапка справочника:

1. Id.
2. Code – строка. Идентификатор состава.
3. Description – строка. Описание состава.

Табличная часть «Единицы транспорта»:

1. Id
2. VehicleId – ссылка на справочник «**Транспортные средства**».

Работа со справочником.

* 1. Экран справочника состоит из трех частей:
     1. **Список составов**. Левая сторона экрана. Стандартный перечень функций: сортировка, поиск, постраничный вывод.
     2. Экранная часть – **шапка справочника**. Правая верхняя часть. Поля шапки справочника.
     3. Экранная часть – табличная часть – **единицы транспорта**. Правая нижняя часть.
        1. № п/п.
        2. Тип транспортного средства.
        3. Регистрационный номер.
        4. Название единицы.

Возможность добавить, удалить единицу транспорта. При добавлении единицы транспорта необходимо выводить диалоговое окно с экранной формой справочника «**Транспортные средства**». На этой форме можно отфильтровать, найти, выбрать необходимую единицу и подтвердить выбор.

----

Справочник предназначен для регистрации объектов транспорта, который отслеживается в модуле "**Трекинг**". В транспорте могут содержаться контейнеры, содержимое которых тоже может отслеживаться по маршруту.

Название в базе данных: **Transports**.

### Справочник маршрутов в разрезе транспортных средств

Справочник предназначен для закрепления маршрута за транспортными средствами

Рекомендуемое название в базе данных: **RouteTransportLinks**.

Структура справочника:

1. Id
2. RouteId – ссылка на справочник маршрутов.
3. TrasportId – ссылка на справочник транспорта.

### Справочник типов пунктов маршрута

Справочник предназначен для регистрации типов пунктов маршрута.

Модуль: **Трекинг**.

Название справочника в базе данных: **RouteNodeTypes**.

Меню: **Справочники** модуля **Трекинг**.

Структура справочника:

1. Id
2. Code – строка. Код типа. Уникальный.
3. Description – описание типа пункта маршрута.

Стандартная функциональность справочника.

### Справочник пунктов маршрута

Справочник предназначен для регистрации пунктов маршрута.

Модуль: **Трекинг**.

Название справочника в базе данных: **RouteNodes**.

Меню: **Справочники**.

Структура справочника:

1. Id
2. RouteNodeTypeId - ссылка на тип пункта маршрута.
3. RouteId – ссылка на справочник маршрутов.
4. LocationId – ссылка на справочник локаций.
5. GpsLatitude – широта.
6. GpsLongitude – долгота.
7. Radius – размер пункта маршрута на местности.

Стандартная функциональность справочника.

### Справочник звеньев маршрута

Справочник предназначен для регистрации звеньев маршрута, т.е. связей между пунктами маршрута.

Модуль: **Трекинг**.

Название справочника в базе данных: **RouteNodeLinks**.

Меню: **Справочники** модуля **Трекинг**.

Структура справочника:

1. Id
2. RouteId – ссылка на справочник маршрутов.
3. StartRouteNodeId – начальный пункт движения объекта по звену. Ссылка на справочник пунктов маршрута. Выбор пунктов маршрута, указанного в поле **RouteId**.
4. FinishRouteNodeId – конечный пункт движения объекта по звену. Ссылка на справочник пунктов маршрута. Выбор пунктов маршрута, указанного в поле **RouteId**.

Стандартная функциональность справочников.

## Журналы

### Журнал регистрации местонахождений объектов

Журнал предназначен для регистрации локаций, где был замечен объект.

Модуль: **Трекинг**.

Название журнала в базе данных: **TrackingRegistries**.

Меню: **Трекинг**.

Структура журнала:

1. Id
2. EntityType – тип объекта (ОС, ТМЦ, …)
3. EntityGuidId / EntityIntId – ссылка на справочник объектов. Guid или int – в зависимости от типа значения ключа справочника объекта.
4. Radius – (нужно уточнить. Заводилось для отслеживания транспортировки труб).
5. GpsLatitude – широта, где был замечен объект.
6. GpsLongitude – долгота, где был замечен объект.
7. LocationId – локация, где был замечен объект.
8. RegistrationDateTime – время обнаружения объекта.

Стандартная функциональность справочника.

#### Подписка на события изменения местонахождения объектов

1. Формирование документа перемещения ТМЦ при изменении локации объекта ТМЦ.

Если объект ТМЦ поменял свое местоположение, то формируется документ перемещения ТМЦ. В создаваемом документе заполняются начальная и конечная локации, а также сам объект ТМЦ. Остальные поля заполняются из шаблона документа перемещения ТМЦ, указанном в настройках подписки. Реквизиты, на которые можно наложить условия:

* 1. Основное условие: изменилась локация - текущая локация не равна предыдущей.
  2. Текущая локация равна (не равна) локации ABC.
  3. Объект равен (не равен) объекту ABC.
  4. Объект равен (не равен) группе номенклатуры (и его дочерним группам).

### Журнал регистрации выполнения объектом маршрута

Журнал предназначен для регистрации фактического выполнения объектом маршрута.

Название журнала в базе данных: **RouteProgresses**.

Структура журнала:

1. Id.
2. RouteId – ссылка на справочник маршрутов.
3. StartDateTime – дата и время начала движения объекта по маршруту.
4. FinishDateTime – дата и время завершения движения объекта по маршруту.
5. EntityType/EntityGuidId/EntityIntId – ссылка на справочник объекта (в зависимости от типа объекта).

### Журнал регистрации передвижения объекта по пунктам маршрута

Журнал предназначен для регистрации фактического передвижения объекта по запланированным пунктам маршрута.

Рекомендуемое название журнала в базе данных: **RouteNodeProgresses**.

Структура журнала:

1. Id
2. **NodeId** – ссылка на справочник узлов маршрута.
3. **StartDateTime** – время появления объекта на пункте маршрута.
4. **FinishDateTime** – время убытия объекта из пункта маршрута.
5. **EntityType**/**EntityGuidId**/**EntityIntId** – ссылка на справочник объекта (в зависимости от типа объекта).

## Настройка подписки на события модуля (AS-93)

Jira: <http://jira.abitech.kz:8080/browse/AS-93>

1. Подписка на события.
2. Фильтрация событий.
   1. Объект равен (не равен) образцу. Образец может быть:
      1. Объект основного средства, объект ТМЦ, работник.
      2. Класс объектов основного средства, группа номенклатуры ТМЦ, организационная единица. Группы могут включать (не включать) дочерние группы.
3. Обработчик событий. Например, процедура, которая создает документ перемещения ТМЦ.
4. Параметры для обработчика событий. Например, шаблон документа перемещения ТМЦ, на основе которого создается документ перемещения ТМЦ в обработчике событий.

У пользователя должна быть возможность на пользовательском интерфейсе:

1. Добавить/удалить подписку на событие.
2. Выполнить настройку подписки, например, в виде JSON файла.

### Справочник «События модуля «Трекинг объектов»

Справочник предназначен для заведения и настройки подписок на события модуля. Например, на события изменения локации объектом ТМЦ, или объектом Основных средств (оборудования).

Модуль: **IoT**.

Меню: **IoT**>**Справочники**.

Справочник состоит из шапки записи и табличной части с настройками.

1. Код подписки. Строка. Обязательно к заполнению.
2. Описание подписки. Строка. Обязательно к заполнению.
3. Действие, которое необходимо выполнить, если условия при событии выполняются.
4. Шаблон документа перемещения ТМЦ для создания нового документа.
5. Табличная часть, которая описывает условия подписки:
   1. Тип объекта событии системы. Может быть:
      1. Основное средство.
      2. ТМЦ.
      3. Работник.
   2. Вид условия: «равно» или «не равно».
   3. Объект, который служит базой для сравнения. Может быть типа:
      1. Основное средство.
      2. ТМЦ.
      3. Работник.
   4. Группа объектов, в которую должен входить объект, чтобы событие было обработано.
      1. Класс основных средств.
      2. Группа номенклатуры (+ ее дочерние группы).
      3. Организационное подразделение (+ его дочерние подразделения).

# Модуль "ЭкоМониторинг"

Общая структура сущностей модуля:

[https://app.diagrams.net/#Htvolodi%2Fabitech%2Fmain%2Fecology%2FEcology\_Structure.drawio](https://app.diagrams.net/" \l "Htvolodi%2Fabitech%2Fmain%2Fecology%2FEcology_Structure.drawio)

## Справочник "Площадки контроля выбросов" (использовать справочник локаций)

Справочник предназначен для ведения реестра площадок контроля выбросов в модуле "**Экология**".

Рекомендуемое название на английском языке: **EmissionPlaces**.

*Мобильная версия*: не требуется.

Справочник имеет следующую структуру:

1. Код источника — **Code** — строка. Уникальный.
2. Описание — **Description** – строка. Обязательное значение.
3. Местоположение площадки — **LocationId** – ссылка на справочник "**Локации**".

Функциональность справочника — стандартная.

## Справочник "Источники выбросов"

<http://jira.abitech.kz:8080/browse/AS-31>

Справочник предназначен для ведения реестра источников выбросов в модуле "**Экология**".

Рекомендуемое название на английском языке: **EmissionSources**.

Мобильная версия: не требуется.

Справочник имеет следующую структуру:

1. Код источника — **Code** — строка. Уникальный.
2. Описание — **Description** – строка. Обязательное значение.
3. Ед.оборудования — **EquipmentId** – ссылка на позицию основного средства / единицу оборудования.
4. Номер источника — **SourceNumber** – строка? Возможно можно обойтись Кодом источника.
5. Местонахождение / локация – **LocationId** – ссылка на справочник местонахождений. Обязательно к заполнению.

Функциональность справочника — стандартная.

## Справочник "Загрязняющие вещества"

Jira:

<http://jira.abitech.kz:8080/browse/AS-77>

Справочник предназначен для ведения реестра загрязняющих веществ в модуле "**Экология**".

Рекомендуемое название на английском языке: **Substances**.

Мобильная версия: не требуется.

Справочник имеет следующую структуру:

1. Код вещества — **Code** — строка. Уникальный.
2. Описание — **Description** – строка. Обязательное значение.
3. Источник выбросов – **EmissionSourceId** – ссылка на справочник выбросов загрязняющих веществ.

Функциональность справочника — стандартная.

## Справочник "Параметры загрязняющих веществ"

Jira:

<http://jira.abitech.kz:8080/browse/AS-78>

Справочник предназначен для ведения видов параметров загрязняющих веществ в модуле "**Экология**".

Рекомендуемое название на английском языке: **PollutantParams**.

Справочник имеет следующую структуру:

1. Код параметра — **Code** — строка. Уникальный.
2. Описание — **Description** – строка. Обязательное значение.
3. Загрязняющее вещество — **PollutantId** – ссылка на запись таблицы "**Загрязняющее вещество**". Обязательно к заполнению.
4. Единица измерения — **UoMId** – ссылка на запись таблицы "**Единица измерения**". Обязательно к заполнению.
5. Датчик - **IODeviceId** – ссылка на справочник датчиков (IODevices/IoT модуль). Датчик, который обеспечивает показания по этому параметру загрязнений.

Функциональность справочника — стандартная.

## Справочник "Нормативы допустимых выбросов"

Jira:

<http://jira.abitech.kz:8080/browse/AS-79>

Справочник предназначен для ведения величин нормативов для выбросов в модуле "**Экология**".

Рекомендуемое название на английском языке: **EmissionStandards**.

Мобильная версия: не требуется.

Справочник имеет следующую структуру:

1. Код параметра — **Code** — строка. Уникальный.
2. Описание — **Description** – строка. Обязательное значение.
3. Вид загрязняющего вещества — **PollutantId** – ссылка на справочник "**Загрязняющие вещества**".
4. Величина нормы — **Value** – число с плавающей точкой. Обязательно к заполнению.
5. Единица измерения — **UoMId** – ссылка на запись справочника "**Единицы измерения**". Обязательно к заполнению.

Функциональность справочника — стандартная.

## Экранная форма контроля выбросов

Jira:

<http://jira.abitech.kz:8080/browse/AS-76>

## Журнал контроля выбросов для оператора №208

Журнал предназначен для регистрации и контроля текущих выбросов, а также контроля соблюдения норм выбросов.

Рекомендуемое название на английском языке: **CustomerReg208**.

Мобильная версия: не требуется.

Структура журнала:

1. Id
2. Время измерений — ReadingTime – Дата + Время измерения величин выбросов.

## Журнал контроля выбросов для инспекции №208

## Журнал превышения норм выбросов №208 / Уведомления о превышениях нормативов эмиссии по данным автоматизированной системы мониторинга

# Настройка рабочего процесса и отслеживания его выполнения

## Справочники

### Справочник типов документов DocTypes

Jira:

<https://ascoa.atlassian.net/browse/AB-77>

<http://jira.abitech.kz:8080/browse/AS-71>

Модуль “**Управление рабочим потоком/DocWorkflow**”.

В зависимости от назначения документов, они делятся по типам. Для этого ведется справочник типов документов. Структура:

1. **Id** - Guid/int?
2. **Code** – код типа документа.
3. **Description** – описание типа документа.
4. **DefaultWorkflow** - ссылка на таблицу **WorkflowTypes**. При создании нового документа этого типа ему прописывается этот рабочий маршрут по умолчанию.

### Справочник статусов документа DocStatuses

Jira:

<https://ascoa.atlassian.net/browse/AB-79>

<http://jira.abitech.kz:8080/browse/AS-72>

Справочник предназначен для определения видов состояния документа по мере того, как документ движется по рабочему маршруту. Например: «Создан», «К исполнению», «Закрыт», …

Поля документа:

* 1. **Code** – код статуса. Текстовое поле. Обязательно к заполнению.
  2. **Description** – описание статуса. Обязательно к заполнению.

Экранная форма справочника – стандартная форма справочника.

### Справочник вида маршрута документа WorkflowsTypes

Jira:

<https://ascoa.atlassian.net/browse/AB-78>

Справочник предназначен для определения маршрута движения документа «Наряд-задание». Маршрут движения – это последовательность пунктов согласований, работ с документом, которые проходит документ в течение своего жизненного цикла. Например:

* + 1. Создан.
    2. Согласован планировщиком.
    3. Согласован начальником цеха.
    4. В работе.
    5. Завершен.
    6. Принят.
    7. Закрыт.

Для определения маршрута документа используется поле «**Вид маршрута**». При создании наряд-задании без использования шаблонов в это поле прописывается вид маршрута по умолчанию. Вид маршрута по умолчанию определяется по типу документа из справочника типов наряд-заданий (**JobTypes**), поле **DefaultWorkflowId**.

Структура справочника **WorkflowTypes**:

1. **Code** – **код** типа. Обязательное к заполнению. Некоторые записи могут быть системными – пользователь не может удалить или изменить.
2. **Description** – **описание** типа. Обязательное к заполнению.

Системные коды:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| DRAFT | Черновик |
| APPROVED | Утвержден |
| IN\_PROGRESS | В работе |
| FINISHED | Работа закончена |
| CLOSED | Документ закрыт |
| CANCELED | Отменен |
| POSTPONED | Отложен |

Экранная форма справочник включает в себя несколько табличных частей:

Табличная часть справочника «***Переходы между статусами документа***» (**WorkflowActions)**:

* 1. **CurrentStatus** – текущий статус – пусто или ссылка на запись справочника **WorkStatuses**. Пусто, если это начало маршрута. Для вида маршрута может быть только одна запись с пустым началом маршрута, так как иначе система не сможет определить начальный статус однозначно.
  2. **NextStatus** — следующий статус- пусто или ссылка на запись справочника **WorkStatuses**. Пусто, если это конец маршрута.
  3. Табличная часть «Роли» ***WorkflowActionRoles***:
     1. **RoleId** – роль, которая может выполнить такой переход – ссылка на запись справочника **Роли** в системе.
     2. **IsAlloved** – признак, что роль может переводить документ из текущего статуса в следующий. По умолчанию — ***true***.
  4. Табличная часть «Работники» ***WorkflowActionEmployees***:
     1. **UserId** - Работник, который может выполнить такой переход – ссылка на справочник работников. Может быть пустым.

Пример экранной формы для справочника маршрутов документов:

[https://app.diagrams.net/#Htvolodi%2Fabitech%2Fmain%2FDoc\_Workflow.drawio](https://app.diagrams.net/" \l "Htvolodi%2Fabitech%2Fmain%2FDoc_Workflow.drawio)

### Определение списка статусов документа к переходу (AS-100)

Jira:

<http://jira.abitech.kz:8080/browse/AS-100>

Движение документа по маршруту рабочего процесса выполняется вручную посредством назначения следующего статуса документа, в который пользователь хочет перевести его.

В текущей версии доступны только последовательные переходы между статусами: от одного пользователя к другому, без параллельных обработок, типа группового согласования, голосования и т.д.

Для перехода в следующий статус используется ниспадающее меню на форме документа, в котором отображены те статусы, в которые может перевести из текущего состояния данный пользователь с его полномочиями.

Пример алгоритма формирования такого списка статусов:

[https://app.diagrams.net/#Htvolodi%2Fabitech%2Fmain%2FDoc\_Workflow.drawio](https://app.diagrams.net/" \l "Htvolodi%2Fabitech%2Fmain%2FDoc_Workflow.drawio)



### История движения документа по маршруту рабочего процесса

Jira:

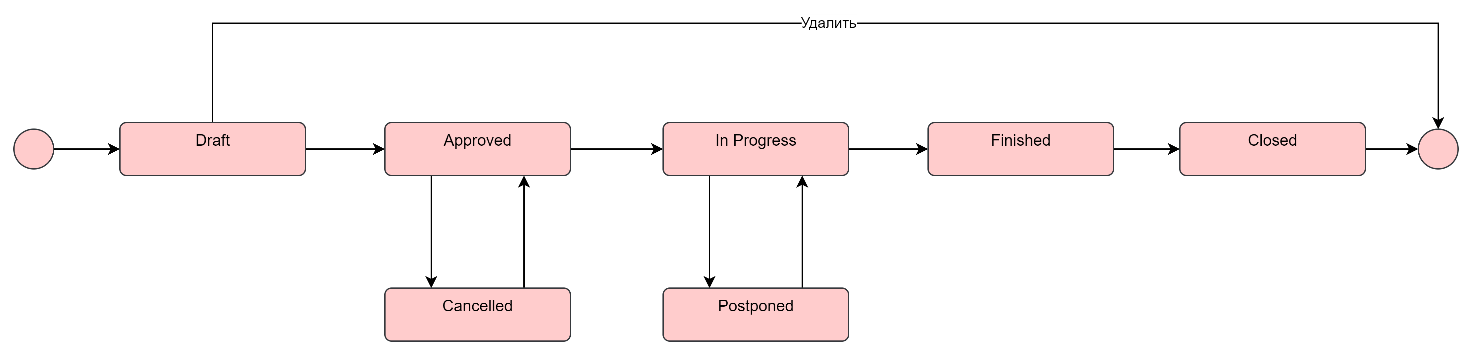
<http://jira.abitech.kz:8080/browse/AS-8>

Для хранения истории выполнения документов используется таблица-журнал «***История выполнения документов*» DocExecLog**.

Структура журнала:

1. **DocId** – ссылка на документ, который двигается по маршруту.
2. **StatusId** – ссылка на статус документа в маршруте рабочего процесса.
3. **StatusStartTime** – время, когда документ был переведен в определенный статус.
4. **StatusFinishTime** – время, когда документ был переведен в следующий статус.
5. **StatusStartUserId** – ссылка на пользователя, кто перевел документ в этот статус.
6. **StatusFinishUserId** – ссылка на пользователя, кто перевел документ в следующий статус.

#### Движение документа по рабочему маршруту



В процессе своего жизненного цикла документ проходит через состояния:

1. **Draft** – черновик. При создании документа ему присваивается статус «Черновик». Документ в этом состоянии подготавливается к движению по рабочему маршруту. В этом состоянии документ может быть сохранен, а также удален. Если документ перевели из этого статуса в другой, то тип рабочего процесс поменять нельзя.
2. **Approved** – утвержден. Перевод документа в этот статус означает, что документ готов к работе и ждет начала выполнения работ по нему. Документ в этом состоянии не может быть удален. Но документ может быть отменен путем перевода его в статус **Cancelled**.
3. **In Progress** – в работе/в процессе. При начале выполнения работы по документу он переводится в статус «**В работе**». Перевод в этот статус может выполняться как вручную, так автоматически, например, при начале заполнения формуляров данных, связанных с оборудованием (фактически начатые работы). Момент перевода документа в этот статус является фактическим началом работы. Документ в этом состоянии не может быть удален или отменен. Он может быть приостановлен или отмечен, как выполнен, если условия работы изменились в процессе работы. Также документ может быть переведен на предыдущие стадии рабочего процесса, например в статус Draft, если необходимо исправить документ. При этом документ нельзя удалить, так как есть история жизни документа.
4. **Finished** – работа завершена. Перевод документа в этот статус означает фактическое завершение работы. В этом состоянии документ может быть изменен только в части формирования подписей документа.
5. **Closed** – документ закрыт. В этом состоянии документ не может изменяться.

Все переходы документа настраиваются для каждого рабочего процесса отдельно. Система не контролирует логичность переходов между состояниями. Это ответственность специалиста, который настраивает рабочий процесс. Таким образом, возможна ситуация, когда документ может быть переведен на предыдущие этапы. Где документ будет откорректирован и переведен в статус закрытого опять.

Перевод документа в следующий статус выполняется:

1. Выбирается новый статус в меню выбора статуса документа.
2. При сохранении документа статус документа изменится на новый.

#### Регистрация истории движения документа по рабочему маршруту (AS-101)

Jira:

<http://jira.abitech.kz:8080/browse/AS-101>

При сохранении документа проверяется, что статус документа изменился. Если статус изменился, то тогда в таблице **DocExecLog** регистрируется новая запись, поля которой заполняются следующим образом:

1. **Id**
2. **DocId** = ссылка на текущий документ.
3. **StatusId** = текущий (новый) статус документ – ссылка на статус маршрута рабочего процесса.
4. **StatusStartTime** = текущая дата + время.
5. **StatusStartUserId** = текущий пользователь – ссылка на пользователя, который перевел документ в новый статус.

Поля **StatusFinishTime** и **StatusFinishUserId** пока остаются пустыми. Проектировались для заполнения и формирования отчета, какое время документ находился в определенном статусе для проведения хронометража рабочего процесса.

### Регистр номеров документов (AS-19)

Jira:

<http://jira.abitech.kz:8080/browse/AS-19>

Регистр используется для хранения номеров документов в разрезе:

* Оргединицы;
* Типа документа;
* Типа рабочего процесса.

Модуль – **Рабочий процесс/Workflow**.

Меню:

1. En: Directories > Document Numbers.
2. Ru: Справочники > Номера документов.

Название таблицы: **DocNumbers**.

Структура:

1. **Id** – целое – записи заводятся в режиме онлайн.
2. **OrgUnitId** – организационная единица. Ссылка на справочник организационных единиц. Inventory/OrgUnits.
3. **DocTypeId** – тип документа рабочего процесса. Ссылка на DocTypes.
4. **WorkflowTypeId** – тип рабочего процесса. Ссылка на справочник типов рабочего процесса WorkflowsTypes.
5. **DocNumber** – целое - текущий номер документа, №п/п.
6. **DocPrefix** – строка – префикс документа.
7. **DocSuffix** – строка – суффикс документа.

Функциональность пользовательского интерфейса – стандартная с возможностью редактирования записей.

Примечание: при создании документа значение DocNumber увеличивается на 1. Необходимо использовать функцию увеличения, которая блокирует запись на момент операции таким образом, чтобы два разных пользователя не получили один и тот же номер.

<https://stackoverflow.com/questions/4358732/is-incrementing-a-field-in-mysql-atomic>

Значение по набору полей:

* OrgUnitId.
* DocTypeId.
* WorkflowTypeId.

является уникальным.

### Справочник компонентов документа

Jira:

<http://jira.abitech.kz:8080/browse/AS-37>

Справочник предназначен для ведения компонентов формы документа.

Модуль — **Рабочий процесс** — **DocWorkflow**.

Меню:

1. En: Directories → Document Components.
2. Ru: Справочники → Компоненты документов.

Таблица **DocComponentTypes**.

Структура таблицы:

1. **Id** – целое — ключ записи.
2. **Code** – строка — код для компонента. Уникальное значение.
3. **Description** – строка — описание компонента. Обязательное значение.

### Справочник компонентов документа по типам документов

Jira:

<http://jira.abitech.kz:8080/browse/AS-38>

Справочник предназначен для регистрации компонентов документа за типами документов.

Модуль — Рабочий процесс / DocWorkflow.

Меню:

1. En: Directories → Components by document type.
2. Ru: Справочники → Компоненты типа документа.

Таблица **— DocTypeComponents.**

Структура таблицы:

1. **Id** – целое — заполняется в онлайн.
2. **DocTypeId** – тип документа — ссылка на таблицу типов документов.
3. **ComponentTypeId** – тип компонента — ссылка на таблицу типов компонентов.

Комбинация DocTypeId + ComponentTypeId является уникальной.

Справочник является табличной частью для справочника «**Типы документов**».

## Порядок работы со статусами документа при движении по маршруту

При создании документа, в справочнике переходов между статусами ищется запись, у которой начальный статус пустой. В справочнике переходов такая запись должна быть единственной. Документу присваивается конечный статус перехода.

При работе с документом пользователь может выбрать следующий статус документа. После сохранения документа документу присваивается этот статус.

Документ заканчивает свой жизненный цикл в статусе, для которого нет следующего статуса для перехода.

При каждом переходе в таблицу Истории выполнения маршрута записывается:

* 1. Информация о новом статусе.
  2. Информация о предыдущем статусе – время выхода из этого статуса и пользователь, который вывел. Для этого ищется запись для этого документа, которая соответствует переходу в этот статус.

## Личный кабинет пользователя

Для группировки документов, где требуется участие пользователя, используется личный кабинет пользователя.

## Доступ к полномочиям системы на уровне документов

Доступ к работе с документом определяется на двух уровнях:

1. Уровень стандартных полномочий для ролей, указанных на уровне системы.
2. Уровень полномочий на уровне статуса документа и роли пользователя, которую он выполняет в процессе.

Для реализации второго уровня задаются роли и/или конкретные пользователи, которые имеют доступ к изменению документа в определенном статусе. При этом учитываются также функции, которые пользователь выполняет при этом: производитель работ, работник.

# Бизнес-кейс: Отслеживание выполнения маршрута ж/д составами, учет пробегов и перевезенных грузов

*Бизнес-кейс рекомендуется выполнять с использованием event-driven architecture.*

Используется функциональность нескольких модулей:

1. IoT – автоматизированный сбор информации о местонахождении объектов подвижного состава.
2. Tracking – формирование и отслеживание маршрута следования подвижного состава. Фиксирование перевозки грузов.
3. Maintenance – формирование типов обслуживания подвижного состава, регистрация состояния оборудования, проведение работ по обслуживанию.
4. Inventory management – формирование движения ТМЦ.

Для регистрации вагонов/локомотивов на пунктах вдоль маршрута следования на вагоне/локомотиве размещается две метки на каждой стороне. Две метки нужны с целью дублирования (резервирования) на случай, если одна из меток перестанет читаться. Тогда у обслуживающего персонала будет время на замену/обслуживание метки, пока вторая метка работает. На каждом пункте маршрута устанавливается две антенны, подключенные к RFID контроллеру. Информация с RFID контроллера передается на сервер в режиме онлайн. Таким образом, антенны привязаны к локации пункта маршрута, и информация об антенне несет информацию о местонахождении вагона/локомотива. Вагон проходит пункт наблюдения, на нем считываются две метки, и в систему передается информация о том, что вагон в такое-то время был на этом пункте.

Контроллер RFID считывает метки со скоростью порядка 250 меток в секунду, при этом подавляющая часть считываний — это повторные считывания одной и той же метки. Такие повторные считывания нужно игнорировать. Эту функцию выполняет встроенное в контроллер программное обеспечение, так называемый встроенный (embedded) драйвер. На выходе из RFID драйвера получаем очищенные данные о чтении RFID меток, нужные для получения информации об объекте наблюдения и для диагностики оборудования: код антенны, метка и время.

Данные передаются на сервер приложений модуля IoT. Здесь данные регистрируются в журнале показаний датчиков и устройств. Регистрируется следующая информация:

1. Код метки.
2. Антенна (привязанная к контроллеру).
3. Время считывания.

По результатам прохождения вагона/локомотива будет две записи для двух меток на вагоне/локомотиве.

Модуль трекинга объектов подписывается на события чтения меток вагонов в модуле IoT. Необходимо одно событие чтения меток: 1-я метка вагона/локомотива (2-я метка вагона/локомотива является дублирующей и служит для контроля работоспособности меток на вагонах/локомотивах). Событие должно нести следующие данные:

1. **Код вагона** — определяется по коду метки на вагоне.
2. **Локация —** определяется по коду антенны, которая привязана к определенной локации.
3. **Время**.

Эта информация регистрируется в Журнале отслеживания / трекинга объекта.

При записи данных в Журнал трекинга объектов запускается процедура, которая рассчитывает пройденное расстояние вагоном/локомотивом после последнего пункта, в котором он был зарегистрирован.

### Определение пройденного расстояния

Пройденное расстояние определяется по справочнику звеньев маршрута. Для этого система ищет звено, конечный пункт которого совпадает с текущим, а первый пункт звена совпадает с предыдущим пунктом, где видели объект. Также ищется звено с обратными пунктами на случай, если объект двигался в обратном направлении.

Если звено не найдено, то формируется запись в журнале показателей оборудования с указанием на ошибку, что звено не найдено. При этом заполняются поля журнала кроме значения пробега. Значение пробега будет восстановлено или при последующей досылке данных от считывателя после устранения проблем, или вручную, если досылка данных невозможна.

При нарушении связи с каким-то пунктом наблюдения объектов данные будут дозаписаны позже, после восстановления связи и выгрузки данных на сервер. Если проблема с самим считывателем, то по результатам выяснения причин оператор вручную дозаписывает соответствующий пробег в журнал показателей пробегов.

Параллельно регистрируется информация в **Журнал выполнения маршрутов**. В нем регистрируется последовательность следования состава через пункты:

Справочник маршрутов (пример заполнения):

|  |  |  |
| --- | --- | --- |
| **Маршрут** | **Пункт** | **Дополнительные признаки** |
| Маршрут 1 | Пункт 1 | Начало / (загрузка ?) |
| Маршрут 1 | Пункт 2 |  |
| Маршрут 1 | Пункт 3 |  |
| Маршрут 1 | Пункт 4 |  |
| Маршрут 1 | Пункт 5 |  |
| Маршрут 1 | Пункт 6 |  |
| Маршрут 1 | Пункт 7 |  |
| Маршрут 1 | Пункт 8 | Конец (выгрузка ?) |
| Маршрут 1 | Пункт 7 |  |
| Маршрут 1 | Пункт 6 |  |
| Маршрут 1 | Пункт 5 |  |
| Маршрут 1 | Пункт 4 |  |
| Маршрут 1 | Пункт 3 |  |
| Маршрут 1 | Пункт 2 |  |
| Маршрут 1 | Пункт 1 | Начало (замкнутый маршрут) |

Справочник звеньев (пример заполнения):

|  |  |  |
| --- | --- | --- |
| **Начальный пункт** | **Конечный пункт** | **Расстояние** |
| Пункт 1 | Пункт 2 | 10 км |
| Пункт 2 | Пункт 3 | 10 км |
| Пункт 3 | Пункт 4 | 10 км |
| Пункт 4 | Пункт 5 | 10 км |
| Пункт 5 | Пункт 6 | 10 км |
| Пункт 6 | Пункт 7 | 10 км |
| Пункт 7 | Пункт 8 | 10 км |
| Пункт 1 | Пункт 2 | 10 км |

Определение пройденного расстояния, если на каком-то пункте не прочитан вагон.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Объект** | **Пункт** | **Время** | **Показатели** |  |
| Вагон 1 | Пункт1 | Время 1 | Предыдущий пункт — "не найден".  Расстояние - 0 |  |
| Вагон 1 | Пункт 2 | Время 2 |  |  |
| Вагон 1 | Пункт 3 | Время 3 |  |  |
| Вагон 1 | Пункт 4 | Время 4 |  |  |
| Вагон 1 | Пункт 5 | Время 5 |  | Не прочиталось |
| Вагон 1 | Пункт 6 | Время 6 |  | Не прочиталось |
| Вагон 1 | Пункт 7 | Время 7 |  |  |
| Вагон 1 | Пункт 8 | Время 8 |  |  |
| Вагон 1 | Пункт 7 | Время 9 |  |  |
| Вагон 1 | Пункт 6 | Время 10 |  |  |
| Вагон 1 | Пункт 5 | Время 11 |  |  |
| Вагон 1 | Пункт 4 | Время 12 |  |  |
| Вагон 1 | Пункт 3 | Время 13 |  |  |
| Вагон 1 | Пункт 2 | Время 14 |  |  |
| Вагон 1 | Пункт 1 | Время 15 |  |  |

после чего передаются в модуль трекинга. Для передачи используется очередь сообщений (Rabbit MQ).

На сервере приложений находится приложение, которое анализирует события чтения меток с целью обнаружения неисправностей оборудования, в частности антенн и меток:

1. Если одна из меток на вагоне не прочитана, то формируется заявка на обслуживание оборудования (метки).
2. Если одна или обе антенны перестали работать, то формируется заявка на обслуживание оборудования (антенна).
3. Если контроллер перестал работать, то формируется заявка на обслуживание оборудования (контроллер RFID).

## Форма начального сопоставления локации и пробега

Форма предназначена для регистрации начального пробега объекта подвижного состава и локации, где находится такой объект. При последующем прохождении пунктов регистрации подвижного состава пробег будет увеличиваться автоматически.

## Модуль трекинга объектов — идентификация вагонов и регистрация местонахождения оборудования

Модуль трекинга объектов используется для:

1. Определения пройденного расстояния вагонами/локомотивами.
2. Определения расстояния, пройденного вагонами с грузом.

Общее расстояние, пройденное вагонами или локомотивами определяется на основе информации по прохождению ими пунктов, на которых считываются RFID метки. Эта информация фиксируется в **Модуле IoT** в виде:

1. Код метки.
2. Локация.
3. Время.

Полученное пройденное расстояние записывается в **Журнал показателей работы оборудования**:

1. Единица оборудования.
2. Тип измерения — пробег. Регистрируется нарастающим итогом. Берется предыдущее значение, добавляется текущий пробег. Регистрируется.
3. Величина показателя — 10 км.
4. Время регистрации.

При записи в Журнал показателей работы оборудования запускается процедура проверки превышения величины пробега нормативного значения. Если превышает, то формируется запись в **Журнале запросов на обслуживание**. На основании этой записи выполняется обслуживание вагона/локомотива.

Процедура проверки превышения нормативного пробега берет предыдущее значение

Для регистрации запрошенных/выполненных работ по обслуживанию необходимо реализовать **Журнал запросов на обслуживание** на основе эксплуатационных показателей:

[https://app.diagrams.net/#Htvolodi%2Fabitech%2Fmain%2F%D0%9E%D0%B1%D1%81%D0%BB%D1%83%D0%B6%D0%B8%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5.%20%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D1%8B.drawio](https://app.diagrams.net/" \l "Htvolodi%2Fabitech%2Fmain%2FОбслуживание. Документы.drawio)

1. Единица оборудования
2. Дата и время запроса на обслуживание. Для автоматической регистрации на основании показаний датчиков или времени - время генерации/активации наряд-задания.
3. Дата и время начала обслуживания - когда наряд-задание переведено в статус “В работе”.
4. Дата и время проведения обслуживания - когда наряд-задание было переведено в состояние “Закрыто”.
5. Вид обслуживания.
6. Класс оборудования.
7. Тип показателя - тип показателя, на основе которого было принято решение о проведении обслуживания (давление, температура, пробег).
8. Ед.измерения. Нужно только для экранной формы в качестве информации.
9. Величина показателя на момент запроса. Заполняется на момент формирования записи в Журнале.
10. Величина показателя на момент начала обслуживания. Регистрируется значение показателя, когда наряд-задание переводится в статус “В работе”.
11. Запрос на обслуживание - ссылка на документ Запрос на обслуживание, если запрос создан на основе этого документа.
12. Рабочее задание - ссылка на наряд-задание на проведение обслуживания.
13. Статус - текущий статус заявки на обслуживание.
    1. “Ожидает обслуживания”.
    2. “Обслуживается”.
    3. “Обслужено”.

Порядок заполнения Журнала:

1. Журнал может заполняться вручную или системой на основе показаний датчиков.
2. Заполнение производится в два этапа:
   1. При возникновении условий для проведения оборудования.
   2. При закрытии наряд-задания на проведение обслуживания, зарегистрированного в этом журнале. <https://ascoa.atlassian.net/browse/AB-53>
3. Система регистрирует запись в Журнале на основе записи в Журнал показателей оборудования. При этом проводится проверка условий для выполнения обслуживания по соответствующим правилам.

При выполнении записи в Журнал показателей оборудования необходимо выполнить процедуру проверки выполнения правил, на основе которых формируются запросы на обслуживание.

Если необходимо выполнить такое обслуживание, то формируется запись в Журнале запросов на обслуживание. При этом заполняются все поля журнала, кроме полей, которые относятся к результату проведения обслуживания:

1. Единица оборудования - ссылка на единицу оборудования, для которой регистрируется запрос.
2. Дата и время запроса на обслуживание - текущее время.
3. Вид обслуживания - при записи в Журнал показателей система проверяла, какой вид обслуживания необходимо провести на основании показателей. Регистрируется ссылка на этот вид обслуживания.
4. Класс оборудования - ссылка на класс оборудования единицы оборудования. Для последующих выборок при планировании.
5. Тип показателя - тип показателя, на основе которого было принято решение о проведении обслуживания (давление, температура, пробег).
6. Ед.измерения. Нужно только для экранной формы в качестве информации.
7. Величина показателя на момент запроса. Заполняется на момент формирования записи в Журнале.
8. Запрос на обслуживание - ссылка на документ Запрос на обслуживание, если запрос создан на основе этого документа.
9. Рабочее задание - ссылка на наряд-задание на проведение обслуживания.
   1. Находится следующее наряд-задание для этого вида обслуживания для этой единицы оборудования. Регистрируется ссылка на него.
   2. Если наряд-задание не найдено, то создается новое согласно правил формирования обслуживания. Регистрируется ссылка на него.
10. Статус - текущий статус заявки на обслуживание.
    1. “Ожидает обслуживания”.

Для отслеживания статуса выполнения запросов на обслуживание необходимо обновлять статус записи Журнала запросов на обслуживание.

1. Когда наряд-задание переводится в статус “В работе”. При этом обновляется значение значение показателя из таблицы показателей, на основе которого было сформирован запрос. Например, значение пробега.
2. Когда наряд-задание переводится в статус “Закрыто”. Обновляется дата и время завершения обслуживания. Для классов оборудования, для которых формируются плановые наряд-задания для профилактического обслуживания, также формируется наряд-задание на следующий период по времени или по пробегу/наработке.