ELMA 365

Low-code ECM

Новая парадигма автоматизации документооборота

О компании ELMA

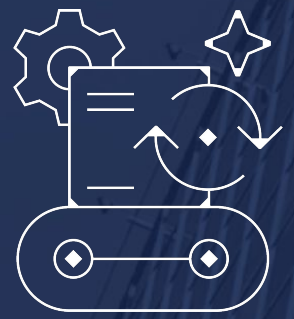
ELMA 365

Более 2000 компаний
используют ELMA

Банк Нейва
ABI PRODUCT
VestaBank
Корпорация Центр
MaximaTelecom
gsk GlaxoSmithKline
РОСГОССТРАХ ЖИЗНЬ
Kapital Bank
Alpen Pharma Group
In the name of health!
АК БАРС БАНК
ПЛЮСБАНК
КуйбышевАзот
DHL EXPRESS
MRIYA RESORT & SPA
AstraZeneca
ancor
КАЛАШНИКОВ КОНЦЕРН
БКС ФИНАНСОВАЯ ГРУППА
ЛокоБанк
УЗТМ УРАЛМАШ
РТ-Инвест
ТЕХНОСТРОМ ГРУППА КОМПАНИЙ
РСУ Управление делами президента РФ
АСЭ·ASE ПРЕДПРИЯТИЕ ГОСКОРПОРАЦИИ «РОСАТОМ»
ЭТП ГПБ Электронная торговая площадка Группа Газпромбанка
МОСКОВСКИЙ КРЕДИТНЫЙ БАНК
SINTEC SINTOIL ОБНИНСКОРГСИНТЕЗ
ТГК-1
МСП Банк
РУССКИЕ БАШНИ

Технологии ELMA365

ELMA 365



Современные технологии
легкие и масштабируемые решения от глобальных ИТ-лидеров

No Microsoft

Golang

ubuntu

PostgreSQL

kubernetes

RabbitMQ

ANGULAR

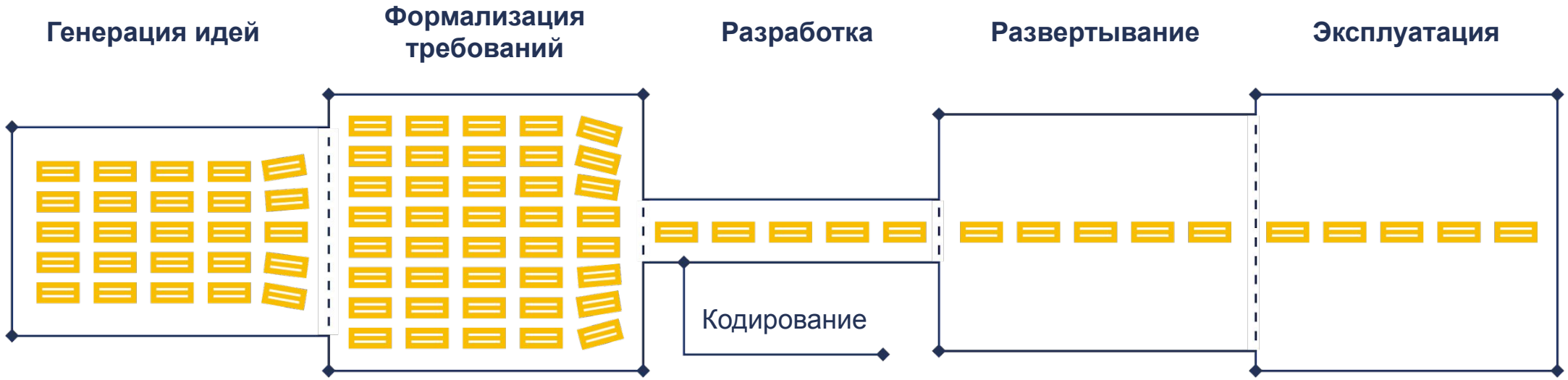
Yandex Cloud

ELMA

Ценность Low-code для бизнеса

Low-code для Бизнеса

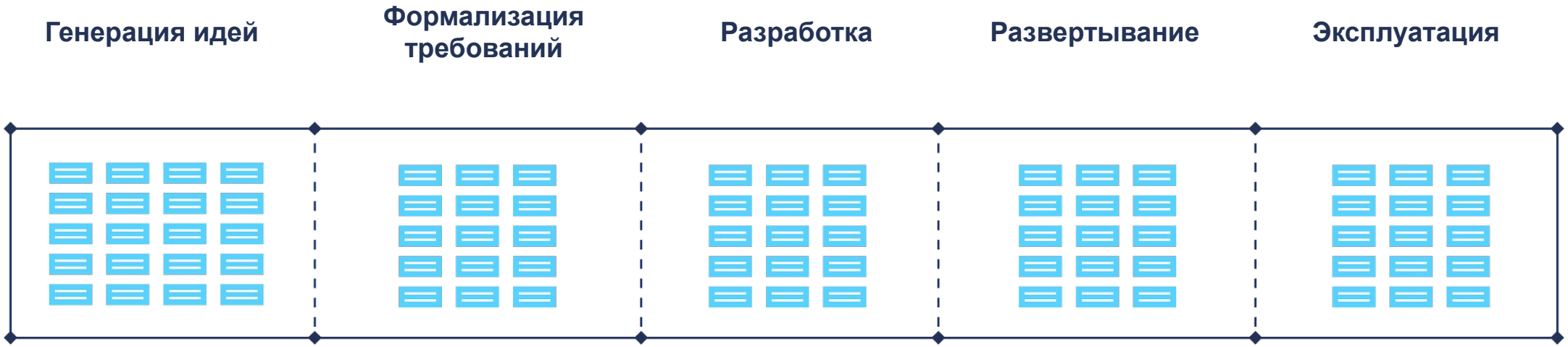
ELMA 365



ТРАДИЦИОННАЯ РАЗРАБОТКА КОРПОРАТИВНЫХ ПРИЛОЖЕНИЙ

Low-code для Бизнеса

ELMA 365

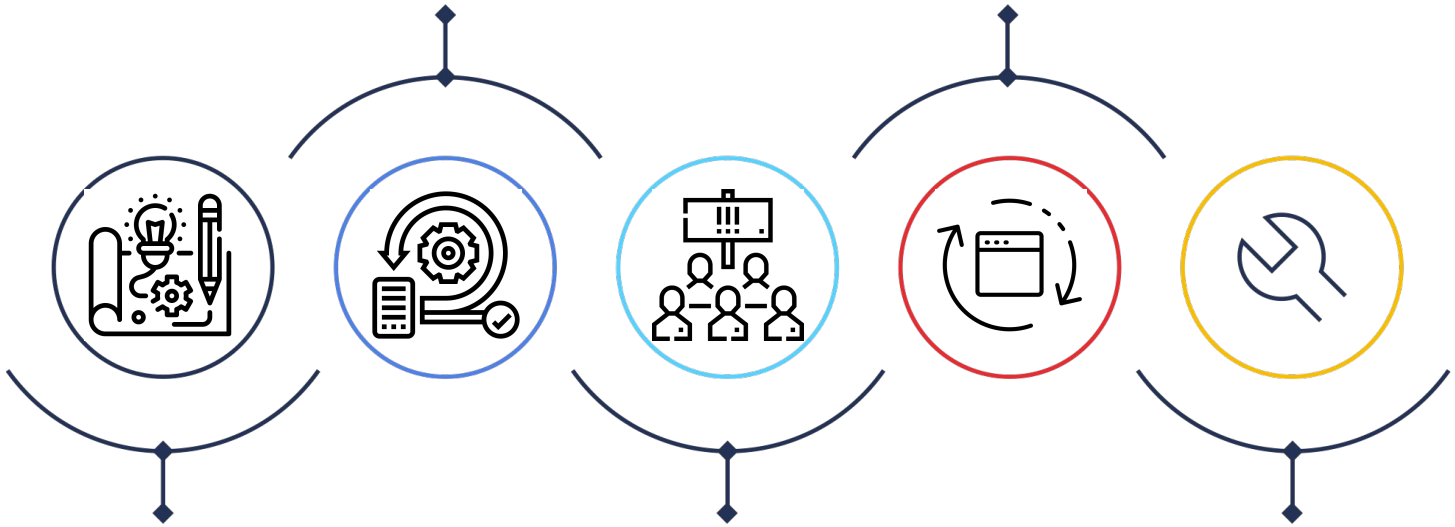


РАЗРАБОТКА LOW-CODE ПРИЛОЖЕНИЙ

Снижение бэклога ИТ создает новое качество разработки, теперь бизнес получает возможность тестировать свои гипотезы в режиме реального времени.

Трансформация процесса разработки

ELMA 365



Low-code меняет процесс разработки, упрощая каждый этап

ELMA

Применение Low-code инструментария при автоматизации документооборота

Контур ЭДО

ELMA 365

Традиционный СЭД

ELMA 365

Документы	Канцелярия	Маршруты	Хранение	...

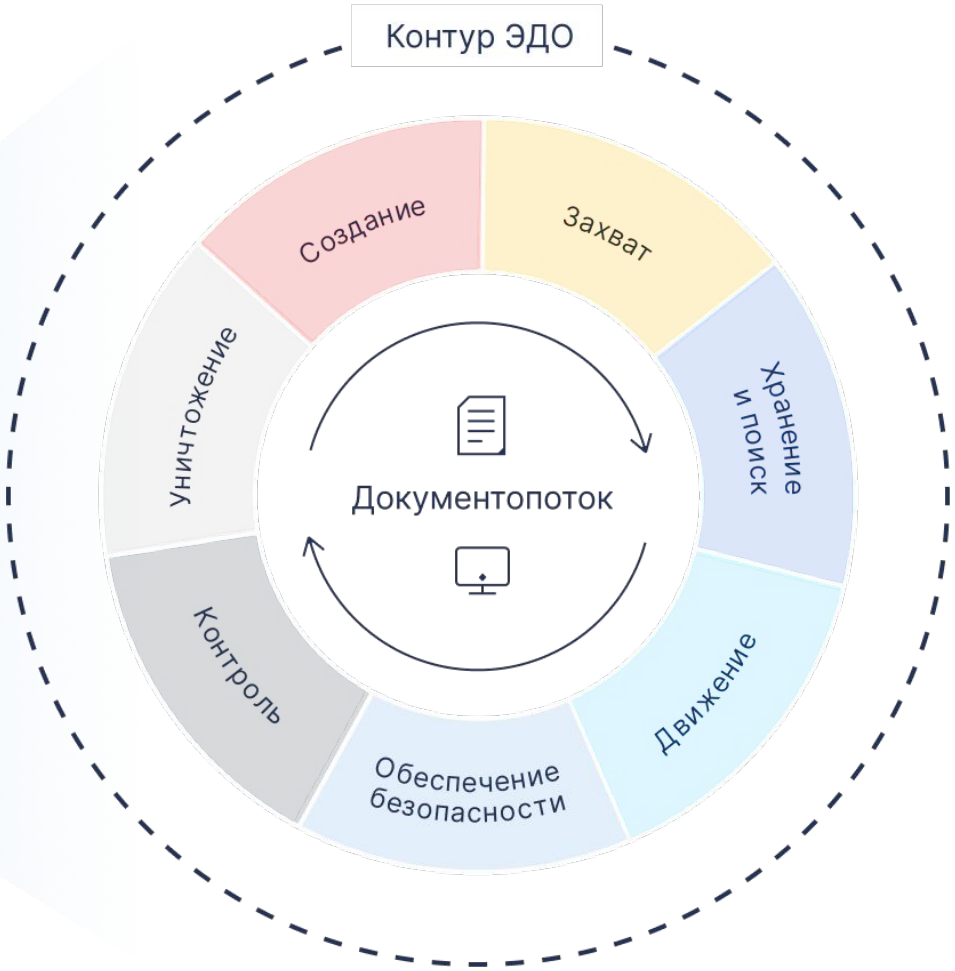
Монолитная система для работы со всеми типами документов

Low-code ECM

ELMA 365

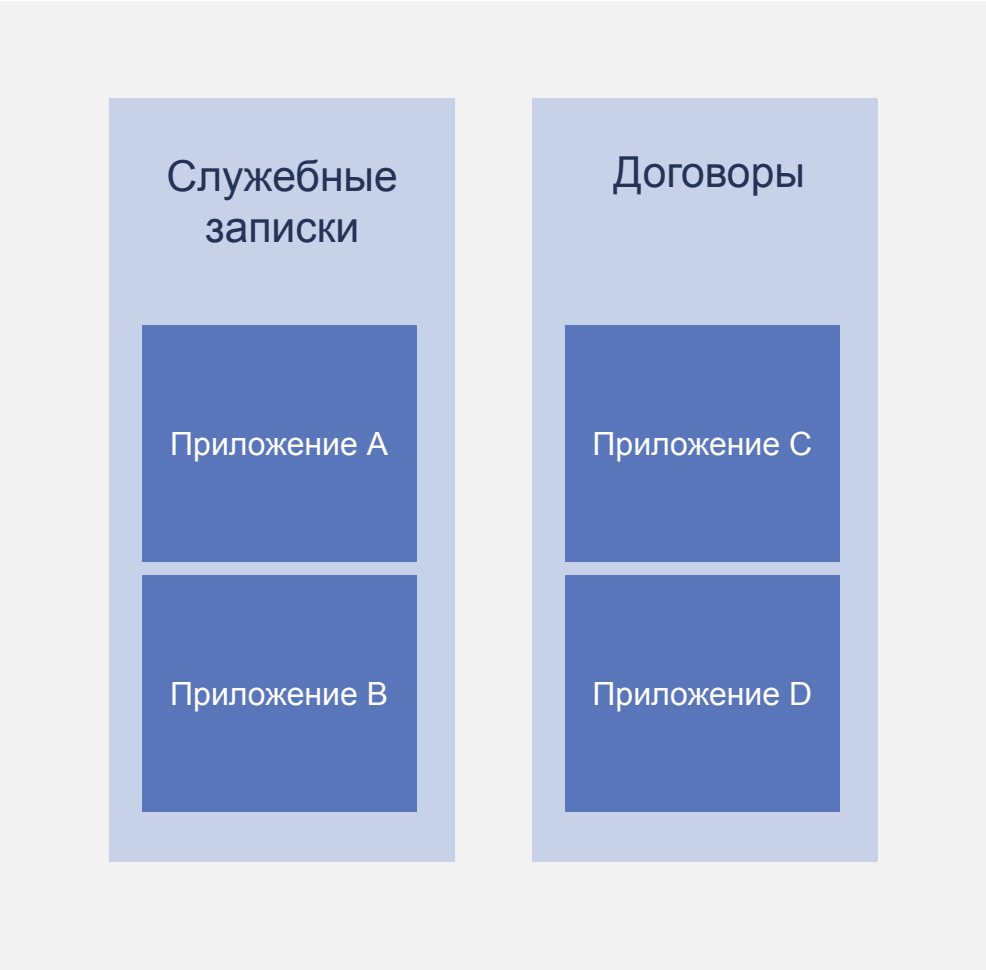
Low-code приложения

- Исходящие договоры
- Входящие договоры
- Письма
- Приказы
- Протоколы совещаний



Изоляция приложений

ELMA 365



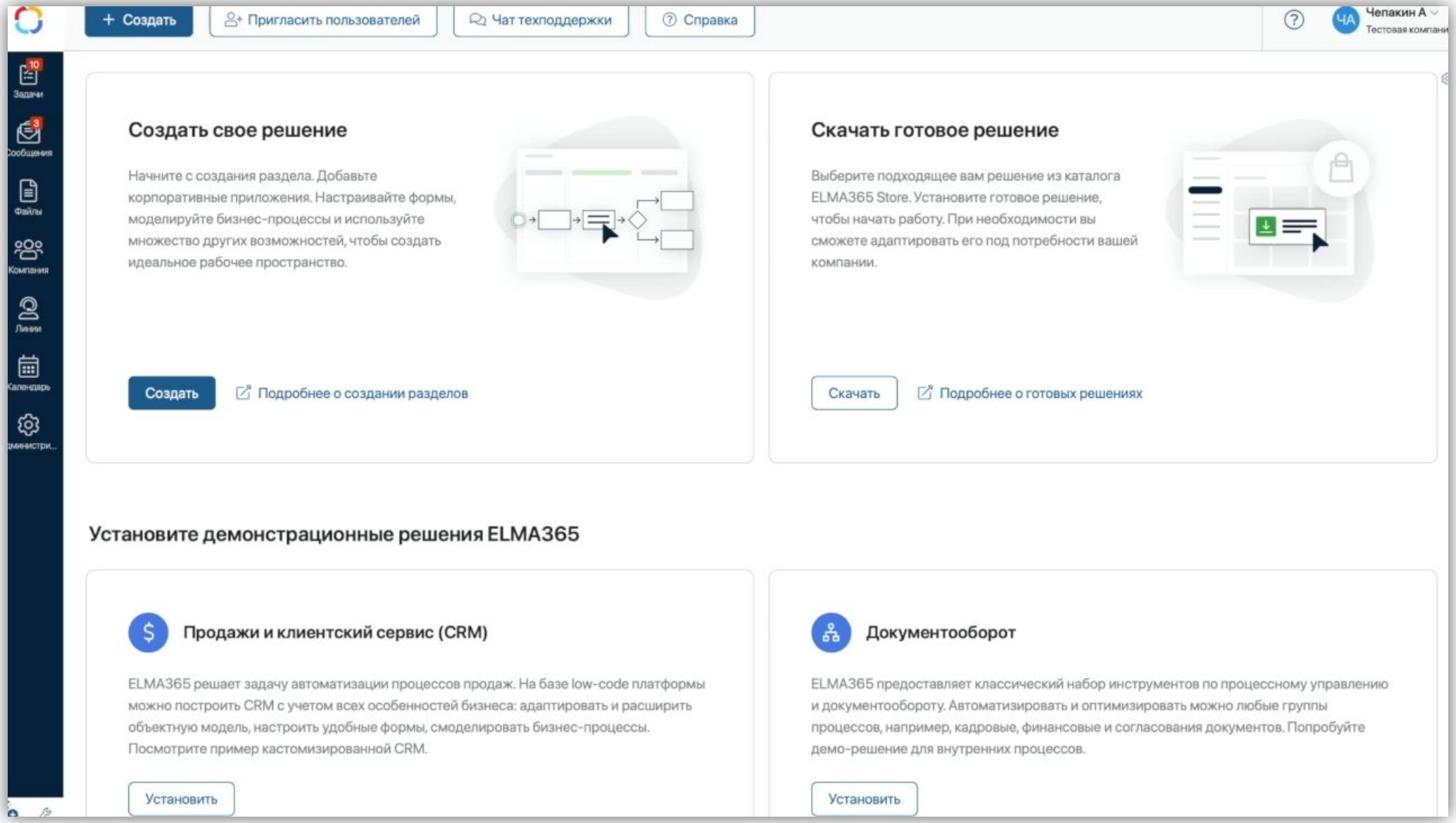
Работа системы 24 на 7, не требуется перезагрузка при внесении каких-либо изменений

Готовые приложения могут быть легко экспортированы и импортированы

Приложения и разделы изолированы друг от друга. Изменения внутри приложения не касаются окружения

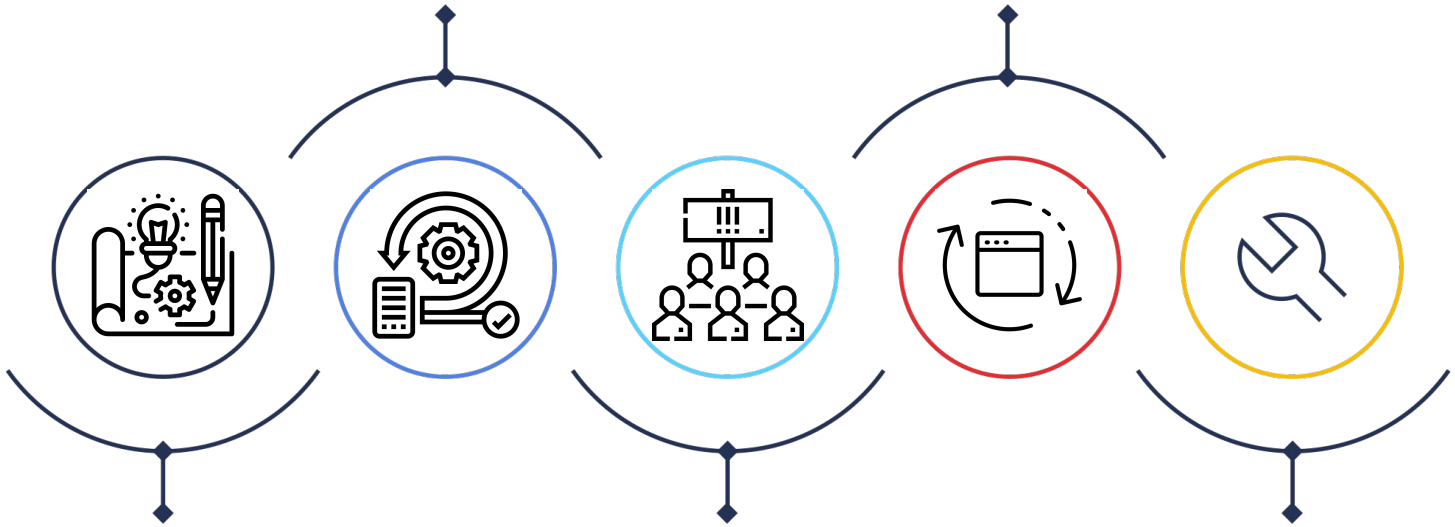
Изоляция приложений

ELMA 365



Разработка Low-code приложения

ELMA 365



Low-code меняет процесс разработки, упрощая каждый этап

Формирование требований

ELMA 365

Формирование требований

Классический подход

Продолжительная и трудоемкая формализация требований

Необходимость оставлять пробелы в требованиях с определением «по ходу проекта»

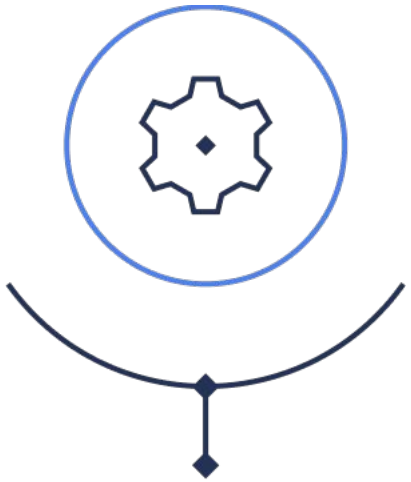
Серьезные требования к аналитикам, работающими над тех. заданием

Low-code

- Требования формируются быстро в лайт–формате
- Подготовка прототипа за несколько дней
- Обсуждение требования на прототипе облегчает понимание Бизнеса и ИТ

Разработка приложения

ELMA 365



Разработка

Классический подход

Высокие требования к команде разработки

Уточнения и пересогласования требований во время разработки увеличивают срок выхода приложения

Ротация команды разработки увеличивает сроки и стоимость

Low-code

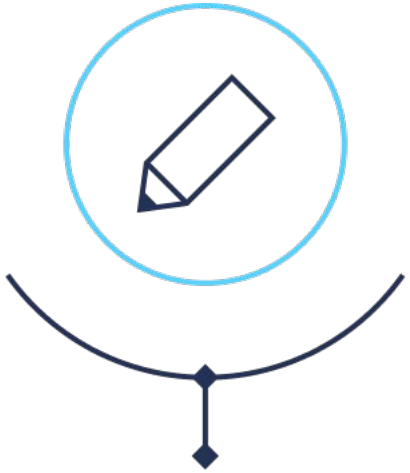
- Создание приложения без кодирования
- Снижение требований к составу команды
- Полное соответствие agile-подходу к разработке – движение спринтами с детализаций требований «по месту»

Прототипирование и разработка

ELMA 365

Стабилизация

ELMA 365



Стабилизация

Классический подход

Устранение замечаний вероятностно приводит к фундаментальной переработке приложения

Изменения требований и тех. задания по инициативе бизнеса в момент демонстрации финального результата

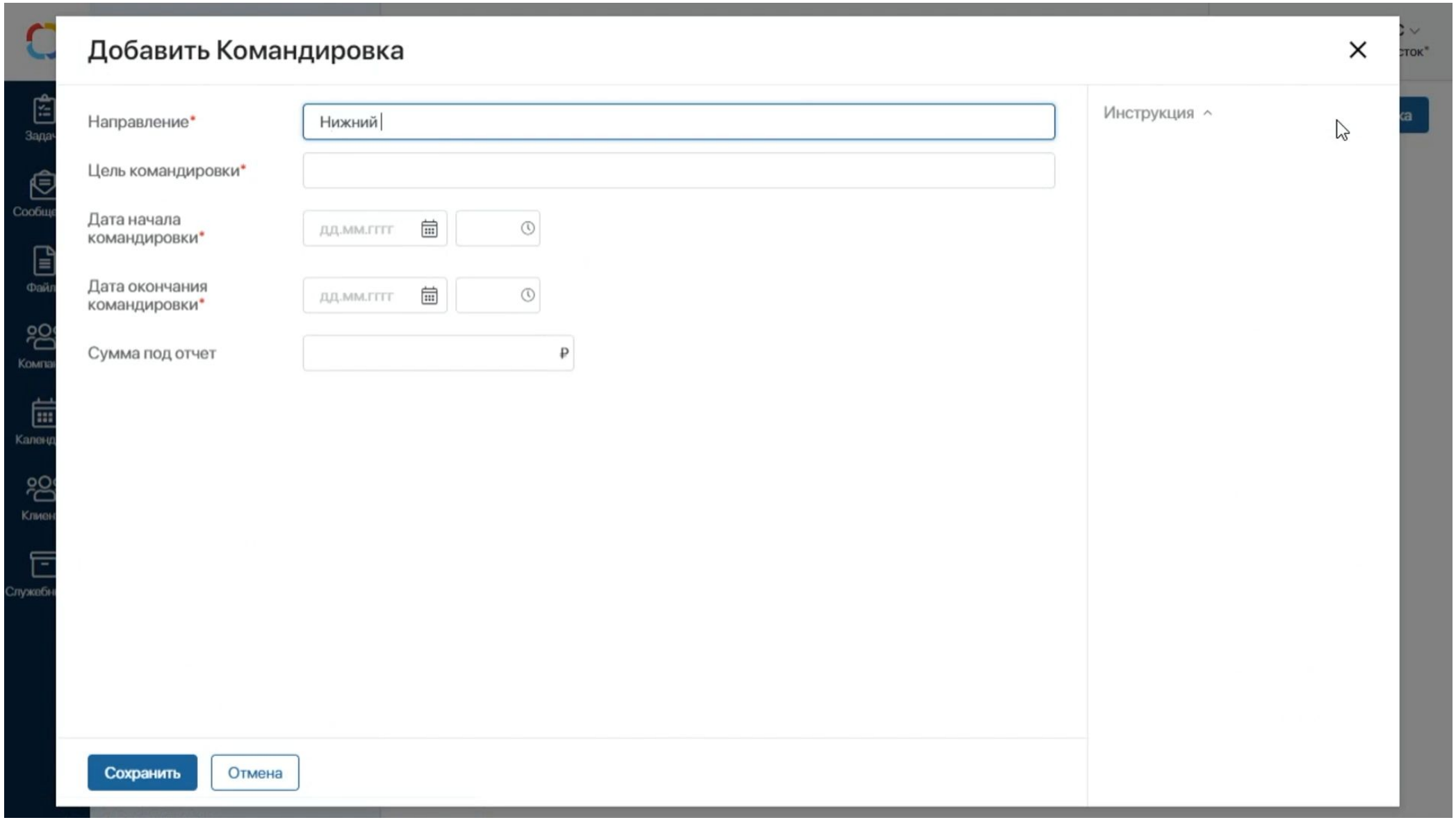
Сложность сдачи интерфейсной части и удобства использования, т.к. эти аспекты часто откладываются «на потом»

Low-code

- Быстрые демо-встречи с бизнес-заказчиками
- Внесение изменений и улучшений «на лету»
- Проработка и адаптация интерфейсов по запросу бизнеса

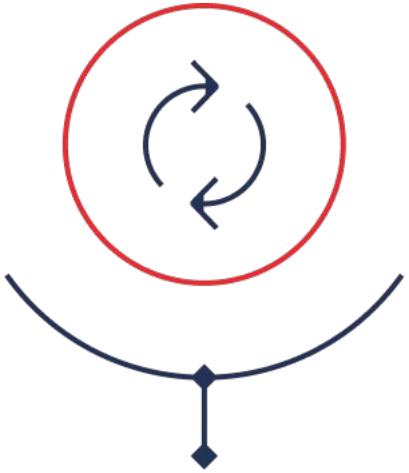
Стабилизация

ELMA 365



Развертывание

ELMA 365



Развертывание

Классический подход

Сложная процедура управления обновлениями и изменениями

Запуск нового приложения порождает риски сопряжения с текущим ИТ-ландшафтом

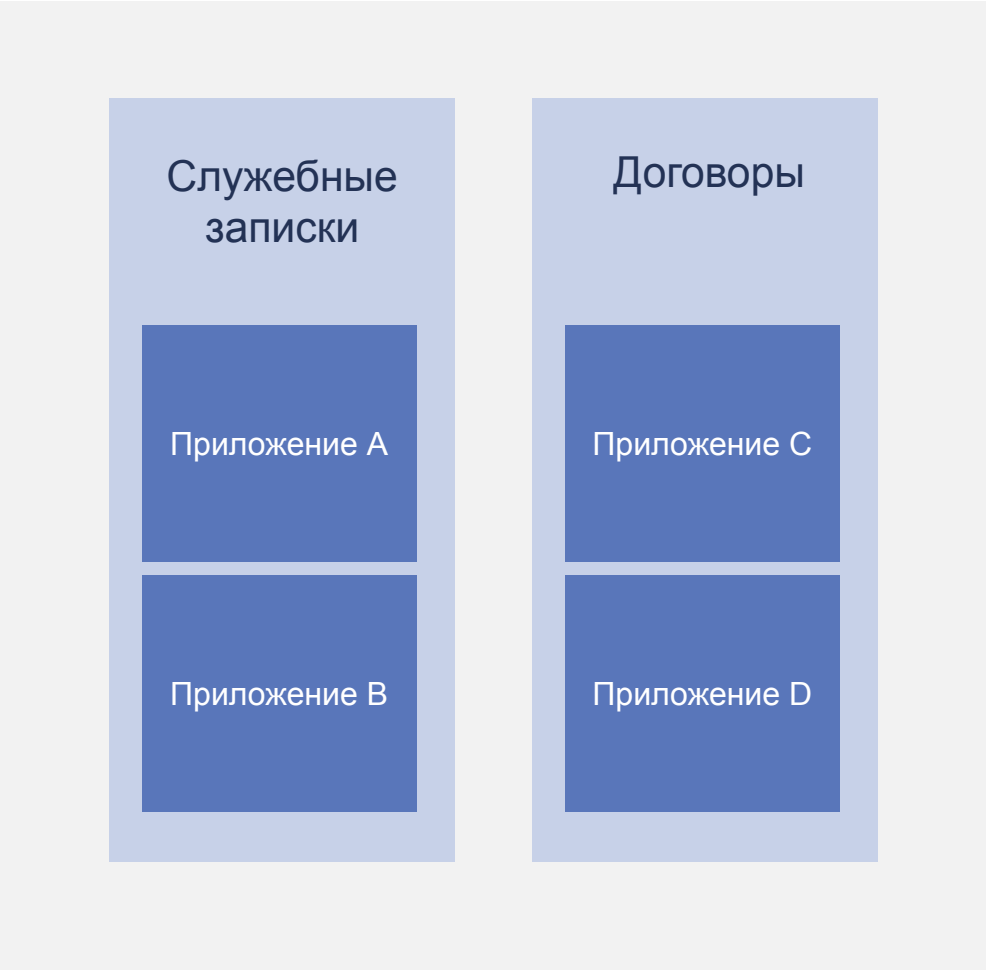
Дополнительные усилия и меры для обеспечения непрерывности работы приложений

Low-code

- Архитектурная изоляция приложений
- Развертывание приложений без остановки системы
- Разделение сред разработки, тестирования и эксплуатации и быстрый перенос приложений между ними

Развертывание

ELMA 365



Работа системы 24 на 7, не требуется перезагрузка при внесении каких-либо изменений

Готовые приложения могут быть легко экспортированы и импортированы

Приложения и разделы изолированы друг от друга. Изменения внутри приложения не касаются окружения

Эксплуатация и поддержка

ELMA 365

Эксплуатация и поддержка

Классический подход

Дополнительные меры и усилия по обеспечению доступности приложений

Значительные расходы на масштабирование при увеличении нагрузки на приложение

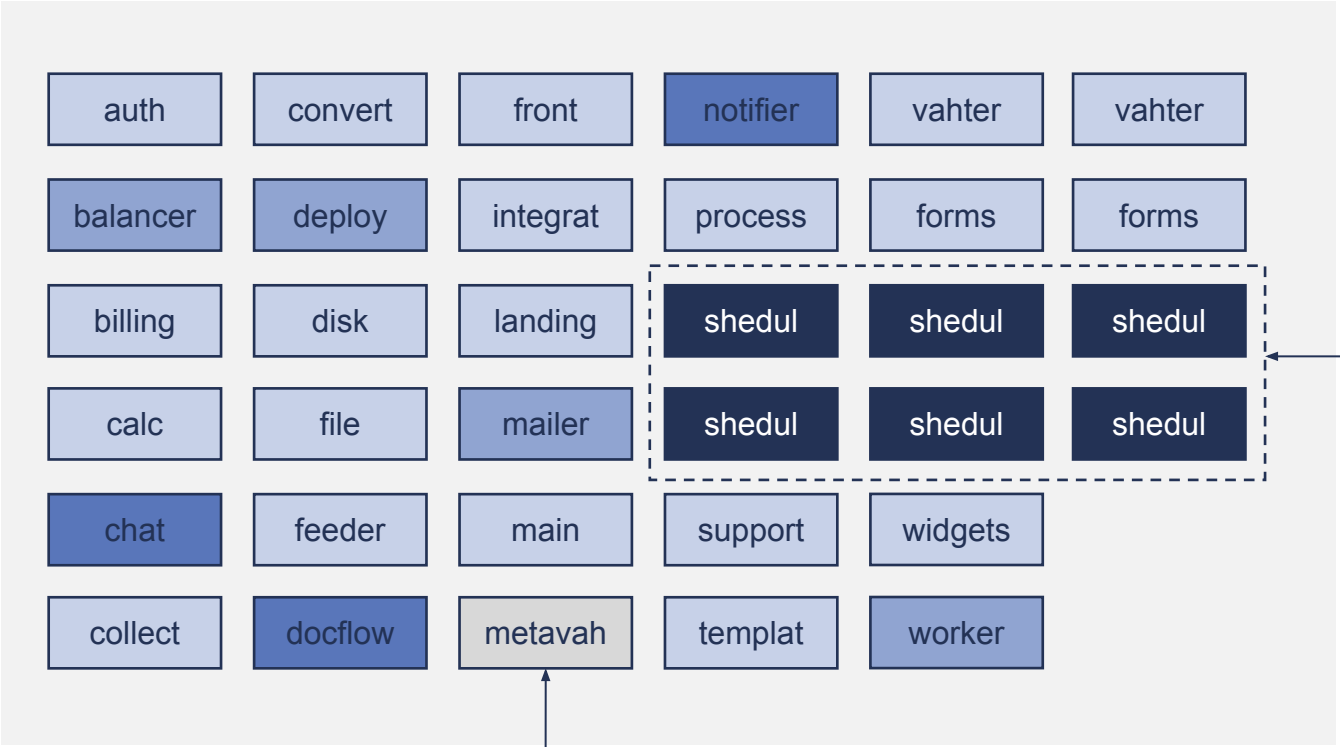
Идеи бизнеса по развитию приложения формируют бэклог в ИТ

Low-code

- Оптимальное использование вычислительных ресурсов
- Быстрое масштабирование мощности при высоких нагрузках
- Неограниченные возможности разработки

Микросервисы, контейнеры, Kubernetes

ELMA 365

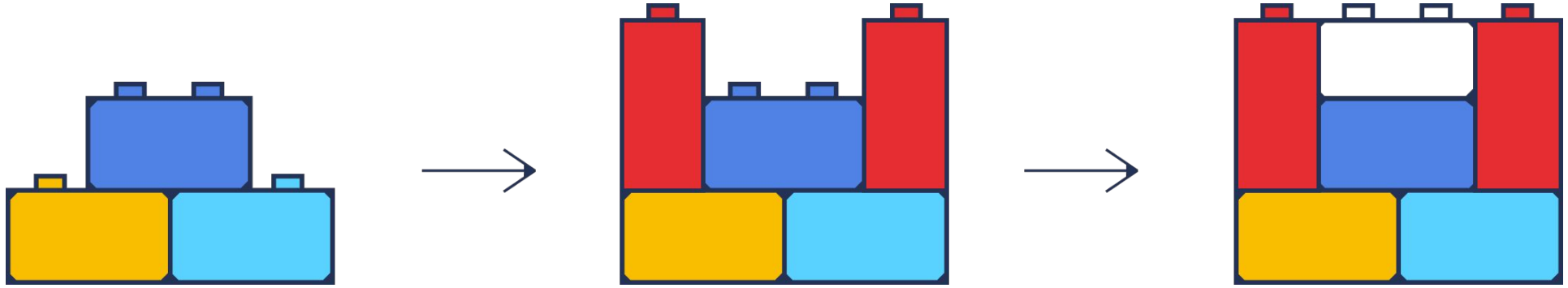


Гибкое масштабирование производительности по высоко-нагруженным микросервисам

Kubernetes. Платформа управления нагрузкой и масштабированием вычислительных мощностей на уровне контейнеров

Принципы: Легкость входа / Многослойность

ELMA 365



ELMA Low-code BPM обеспечивает легкость входа – это значит, что приложение может быть создано быстро и просто. При этом, это приложение может усложняться до необходимой степени с помощью DevOps-инструментария. ELMA Low-code BPM позволяет создавать сколько угодно сложные приложения.

ELMA

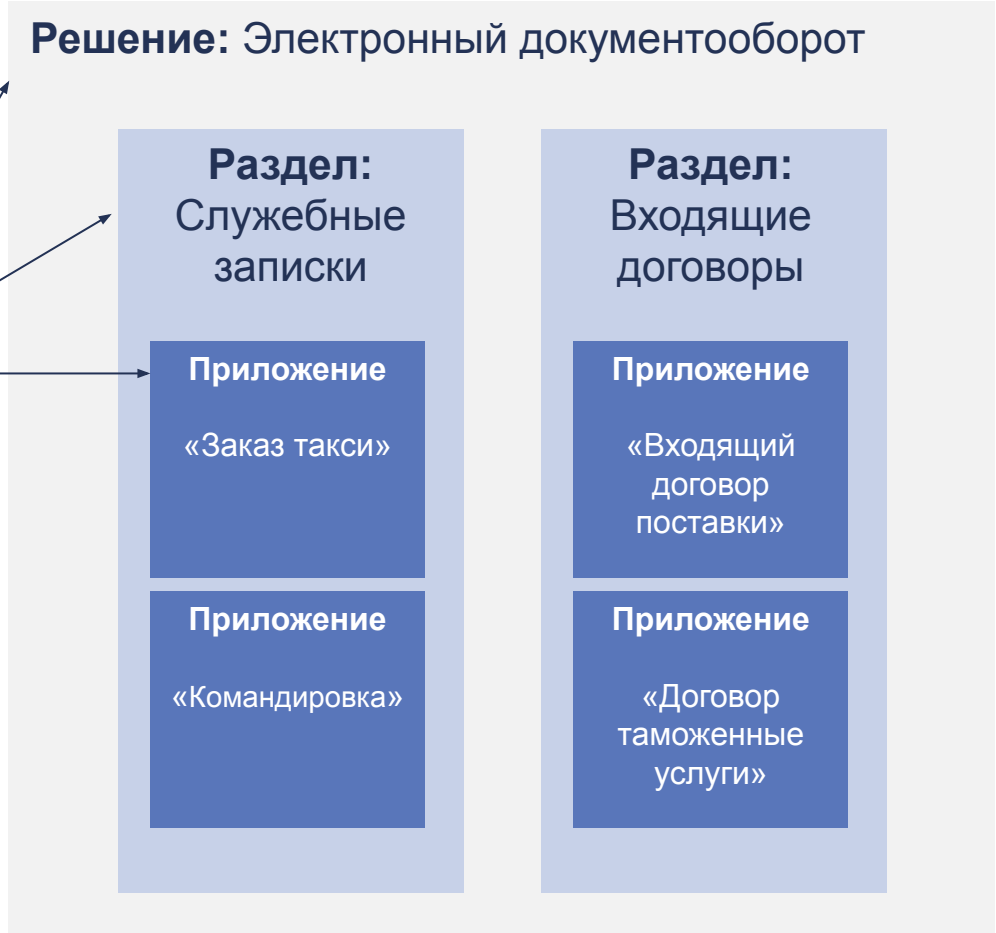
Low-code features

Уровни абстракции и архитектурная изоляция

ELMA 365

Уровни абстракции:

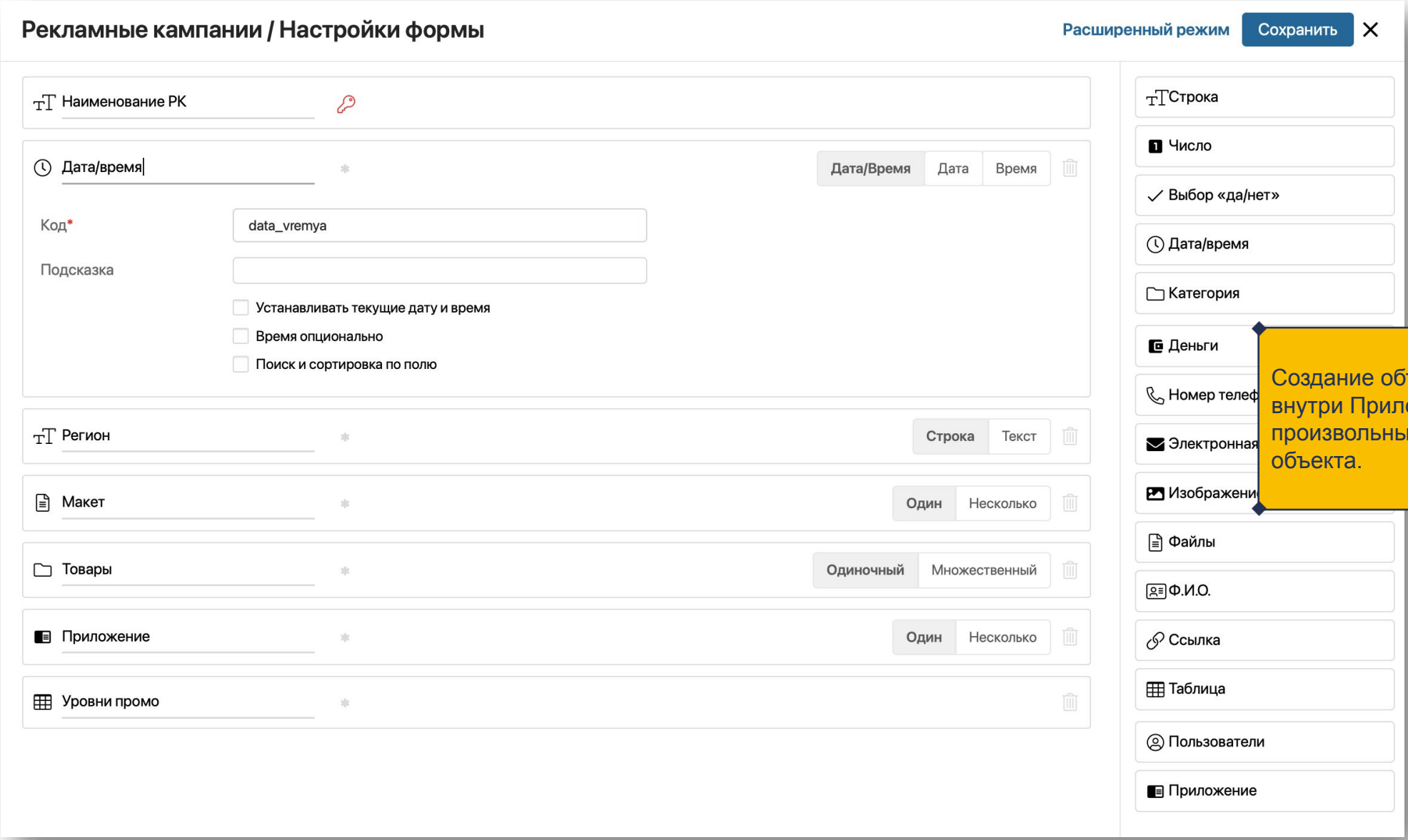
- Решение
- Раздел
- Приложение



- ✔ Обеспечен асинхронный режим разработки и обновления на любом уровне абстракции
- ✔ Принцип независимости от окружения
- ✔ Разработка отдельных приложений, разделов или решений с разными уровнями глубины

Расширяемая объектная модель

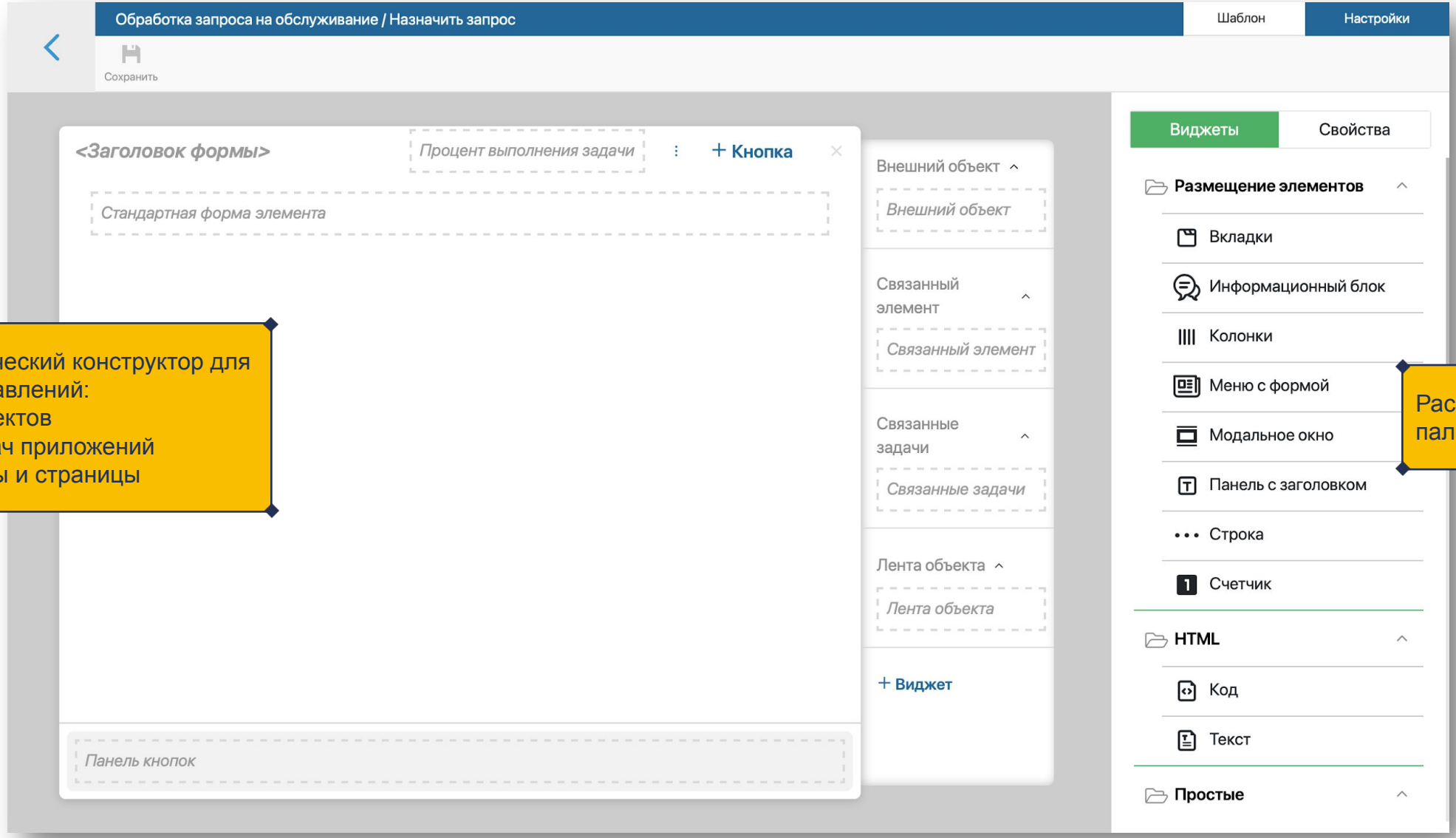
ELMA 365



Создание объекта автоматизации внутри Приложения. Определяем произвольный набор данных для объекта.

Графический конструктор

ELMA 365



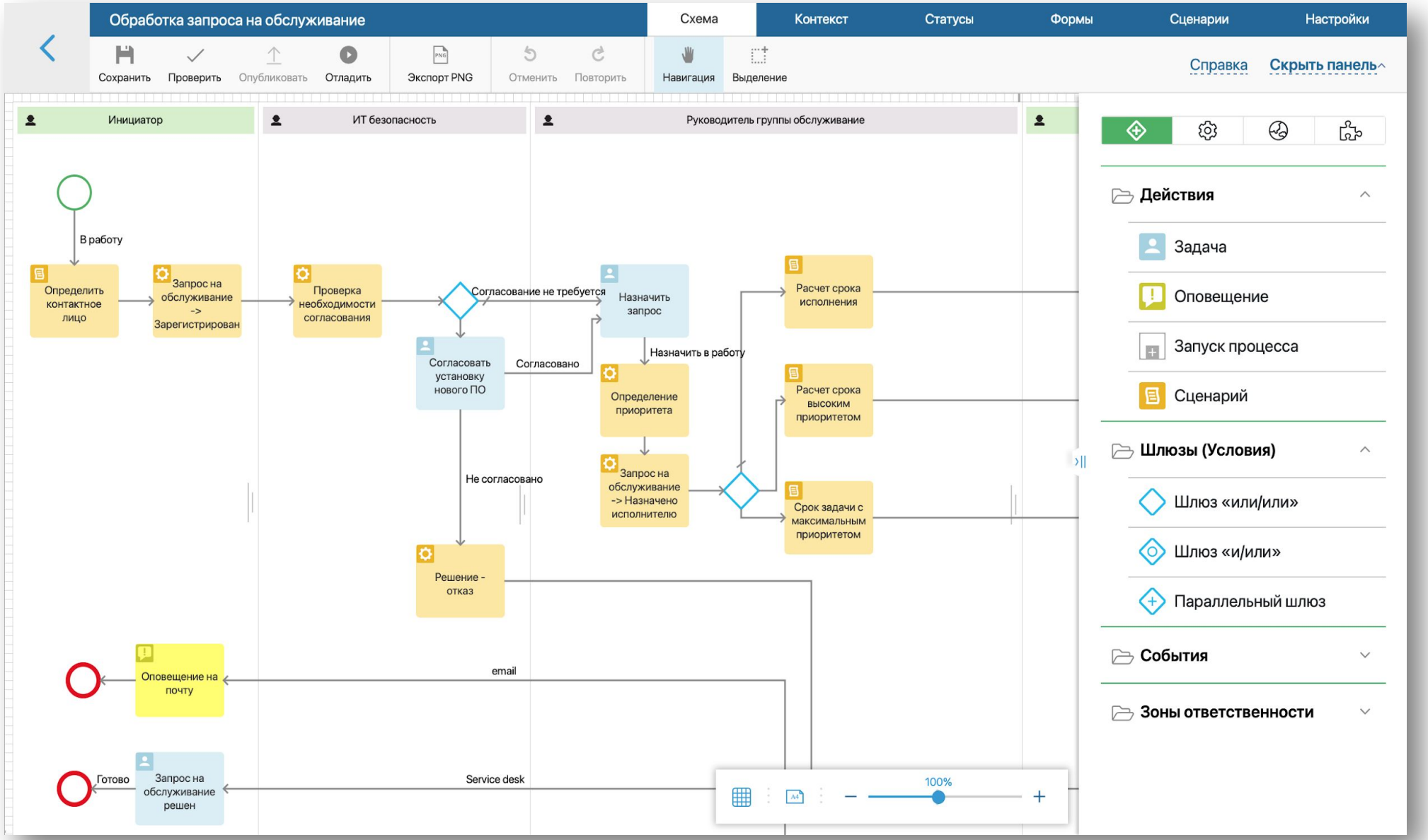
Единый графический конструктор для разных представлений:

1. Формы объектов
2. Формы задач приложений
3. Интерфейсы и страницы

Расширяемая палитра виджетов

BPM для управления бизнес-логикой

ELMA 365



Генерация API для приложений «на лету»

ELMA 365

Service desk

- Главная страница
- Запрос на обслуживан...
- Инцидент
- CMDBtst
- Сервис
- CMDB
- SLA
- Отчет
- Отчет_ демо

Добавить

Настроить

Чепакин А
ООО Тест

Запрос на обслуживание / API

Работа с элементами Приложения осуществляется через Web API. Доступные функции перечислены ниже.
Полезные ссылки: Документация по Web API

Список элементов | Получить элемент | Создать элемент | Изменить элемент | Изменить статус

Описание	Создание элемента Приложения
Метод	post
Адрес	https://o2si34iy66uv6.elma365.ru/pub/v1/app/service_desk_demo/zapros_na_obsluzhivanie/create

(Открыть документацию по методу create)

JSON пакет | Проверить

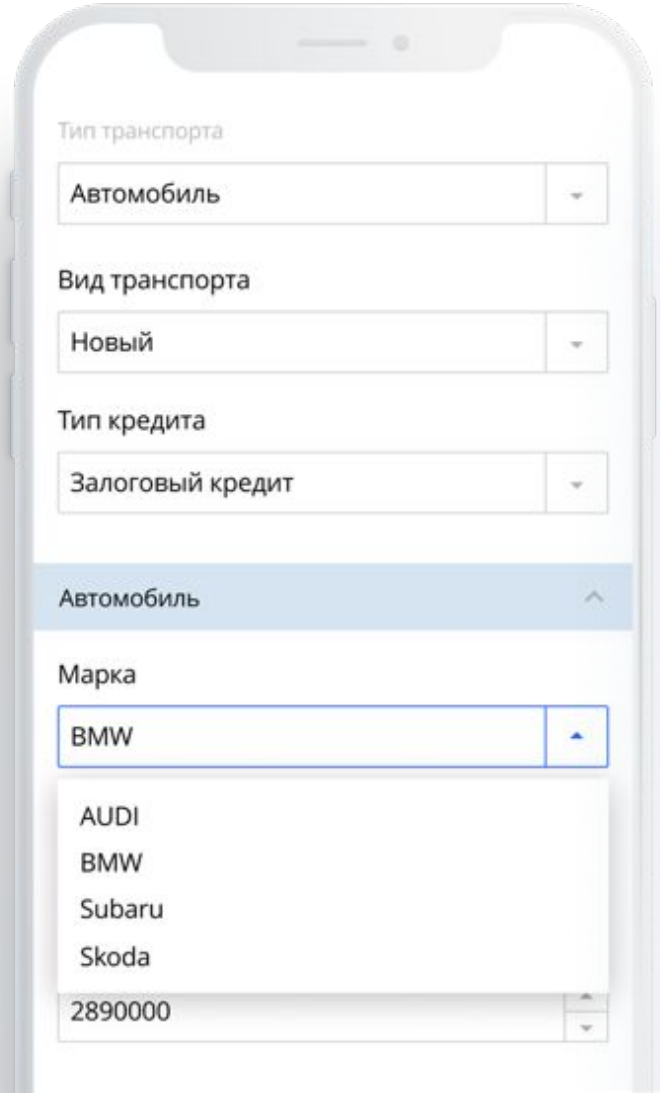

```
{
  "context": {
    "kratkoe_opisanie": "example",
    "opisanie": "example",
    "servis": [
      "00000000-0000-0000-0000-000000000000"
    ],
    "reshenie_predostavit_mne": true,
    "contact_person": [
      "00000000-0000-0000-0000-000000000000"
    ],
    "sposob_obratnoi_svyazi": [
      {
        "code": "code",
        "name": "name"
      }
    ],
    "prioritet": [
      {
        "code": "code",
        "name": "name"
      }
```

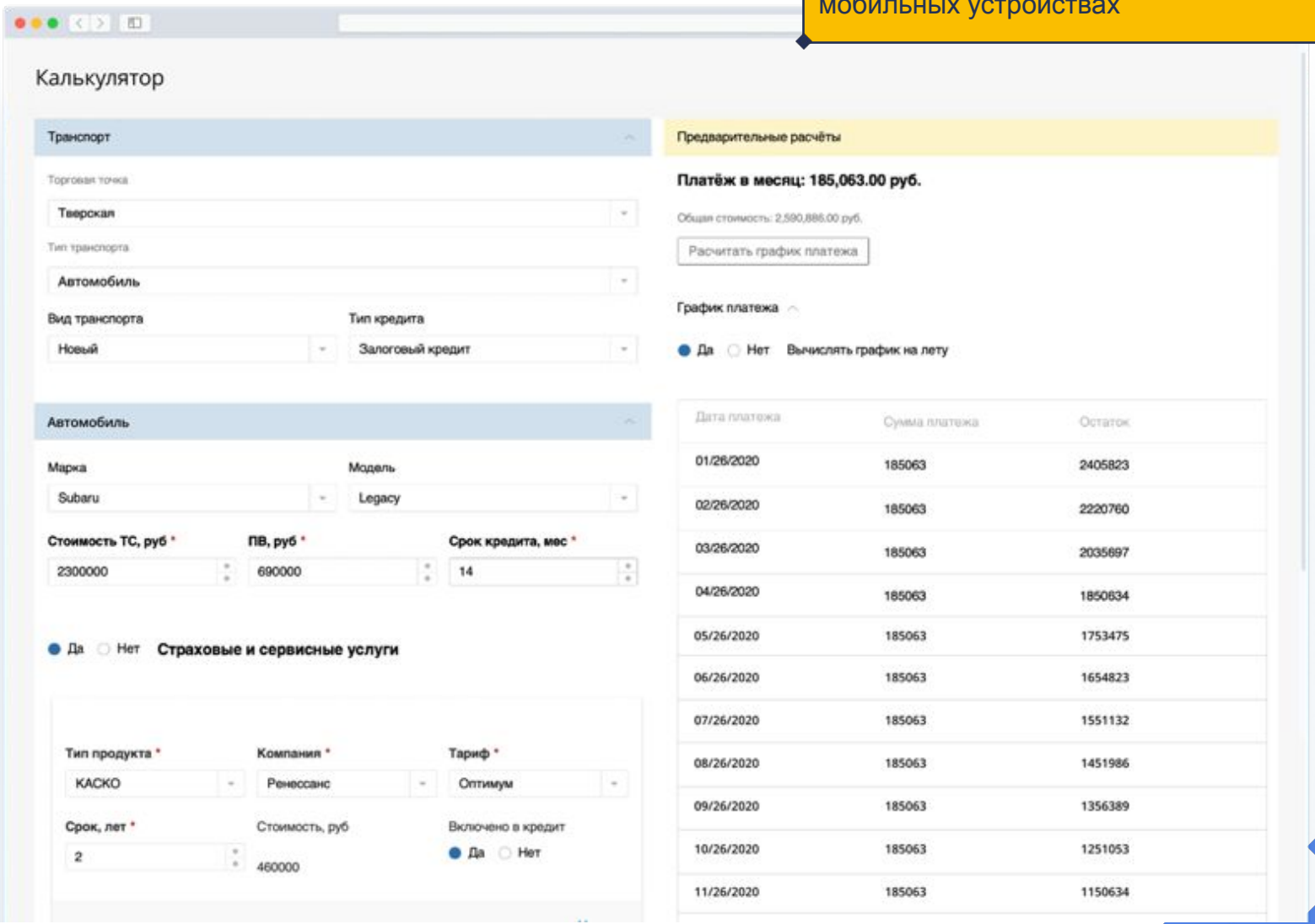

Созданное Приложение автоматически становится доступным для запуска извне веб-сервисом

Мобильное приложение

ELMA 365

Приложение автоматически доступно на мобильных устройствах





Публичное SDK для разработки

ELMA 365

ELMA 365 TS SDK

ELMA365 Справка API

Главная

- Типы объектов

ProcessInstanceState

ProcessTaskState

ApplicationItem

ApplicationItemRef

ApplicationItemRegistration

BaseItem

BaseItemData

CategoryItem

CategoryItemRef

DirectoryData

FileData

FileItem

FileItemRef

ImageData

ImageItem

Главная / Типы объектов

Типы объектов

Объекты системы представляют из себя записи в базе данных. Это могут быть элементы приложений (ApplicationItem) или базовые объекты системы (UserItem, FileItem, ImageItem). Каждый объект описывается набором соответствующих ему полей (ItemData), где коду поля сопоставляется его тип. Также объекты имеют базовый набор полей BaseItemData. Каждый объект содержит описание своих полей (свойство `fields`), а также объект значений этих полей (свойство `data`). Описание поля зависит от его типа.

Ссылки на объекты

Для всех объектов существует также объект-ссылка ItemRef, через который можно получить сам объект. Например, в базовом наборе полей любого объекта есть ссылка на автора этого объекта `__createdBy`, для того, чтобы получить объект автора, необходимо его запросить:

```
const author = await Context.data.__createdBy.fetch();
```

У любой ссылки есть метод `fetch`, с помощью которого можно запросить сам объект.

Пользователи

Пользователь (UserItem) является объектом системы с набором полей UserData. Для ссылки на пользователя используется объект типа UserItemRef. В данный момент пользователей нельзя создавать или изменять из сценариев.

Например, пусть процесс запускается по элементу приложения `order` и необходимо поставить задачу на автора этого элемента, тогда достаточно сделать динамическую зону ответственности и в связанную с ней переменную `author` положить автора элемента:

В этой статье

Ссылки на объекты

Пользователи

Файлы

Элементы приложений

ELMA

Вопросы