

Какие специалисты нужны для реализации Low/No-code проектов?

Максим Тарасов

Руководитель практики систем
управления бизнес-процессами



НОРНИКЕЛЬ

СПУТНИК

Содержание

- **Pro-Code / High-Code** – классическое написание кода на обычных или специализированных языках программирования (JS, C#, Java, Python и др.) *
- **Low-Code** – это методика создания информационных систем, сервисов и программного обеспечения, с минимальным использованием рукописного кода. *
- **No-Code / Zero-Code** – производная от Low Code концепция, предполагающая полное отсутствие рукописного кода в программном обеспечении. *

* - из открытых источников



Стереотипы и факты Low/No-code



Сложности поиска и подбора



Подходы к развитию сотрудников



Выводы

Стереотипы и факты Low/No-code (1\5)

Стереотипы Low/No-code

Универсальность концепции:

- Low-code платформа поможет закрыть любую потребность
- Легкий путь реализации любого проекта

Скорость разработки:

- Гарантирует высокую скорость внесения изменений в ходе проекта\эксплуатации

Сотрудники

- Low-code «разгружает» ИТ
- Большую часть задач закрывают бизнес-аналитики

Быстрота освоения:

- Low-code/No-code платформы практически не требуют обучения

Факты Low/No-code

Универсальность концепции:

- Зависит от масштаба и сложности проекта
- Работает в связке с Pro-code

Скорость разработки:

- Зависит от архитектуры решения и количества кастомного кода

Сотрудники

- В Low-Code командах в основном программисты
- Аналитиков не подпускают к разработке

Быстрота освоения:

- Требуются базовые знания ИТ по ряду направлений

Стереотипы и факты Low/No-code (2\5) Универсальность концепции



No-code

- Разработка прототипа
- Разработка MVP (Minimum Viable Product)
- Визуальный интерфейс



Low-code

- Расширение логики работы No-Code инструментов скриптами
- Использование готовых решений: CRM, ITSM и т.д.



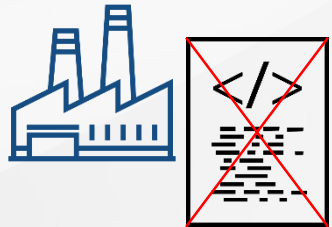
Pro-code

- Реализация сложной логики поведения системы
- Интеграционные коннекторы, не заложенные в базовом пакете

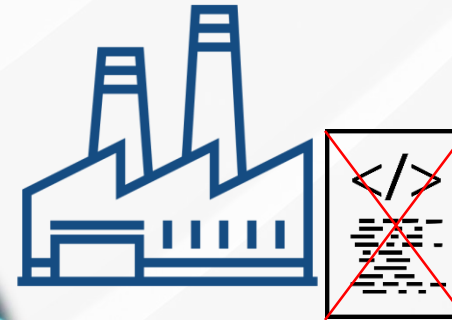
Стереотипы и факты Low/No-code (3\5) Скорость разработки



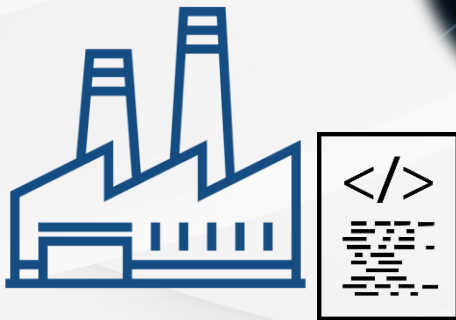
- Небольшая компания, отдел
 - Простые процессы
- ↓
- Low-code\No code



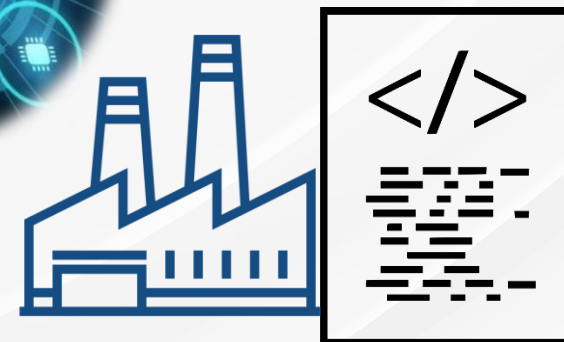
- Большая компания
 - Сложные процессы
- ↓
- Low-code\No code



- Большая компания
 - Сложные процессы
- ↓
- Low-code\Pro code



- Большая компания
 - Сложные процессы
- ↓
- Low-code\много Pro code



Стереотипы и факты Low/No-code (4\5) Сотрудники.



- Не заинтересован погружаться в программирование
- Модели процессов не пригодны для реализации в системе
- Создает и описывает словами и схемами

- Перегружен распланированными на месяцы вперед задачами
- Не подпускает аналитика к Low-code инструменту
- Автоматизирует процессы в системе

Стереотипы и факты Low/No-code (5\5) Быстрота освоения.

	No-code	Low-code	Pro-code
Определение методики	концепция, предполагающая полное отсутствие рукописного кода в программном обеспечении.	методика создания информационных систем, сервисов и программного обеспечения, с минимальным использованием рукописного кода	классическое написание кода на обычных или специализированных языках программирования (JS, C#, Java, Python и др.)
Уровень требований методики к сотрудникам	Базовые знания ИТ	Начинающий разработчик (Junior)	Разработчик среднего уровня (Middle), старший разработчик (Senior)
Время на освоение	Недели	Месяцы	От года, годы

Сложности поиска и подбора

	No-code	Low-code	Pro-code
Определение методики	концепция, предполагающая полное отсутствие рукописного кода в программном обеспечении.	методика создания информационных систем, сервисов и программного обеспечения, с минимальным использованием рукописного кода	классическое написание кода на обычных или специализированных языках программирования (JS, C#, Java, Python и др.)
Уровень требований методики к сотрудникам	Базовые знания ИТ	Начинающий разработчик (Junior)	Разработчик среднего уровня (Middle), старший разработчик (Senior)
Время на освоение	Недели	Месяцы	От года, годы
Вид обучения	Видеокурсы, вебинары, самостоятельное обучение	+ онлайн\оффлайн курсы, техническая документация	+ специальное образование, много практики
Результат	Создание прототипов, проверка гипотез, проведение экспериментов	Реализация процессов с несложной логикой	Сложные группы бизнес-процессов, интеграции, высоконагруженные системы

Подходы к развитию сотрудников

Адаптация

- Комфортная коммуникационная среда с коллегами. Площадки для обмена мнениями (ежедневные стендапы, статусные встречи и т.д.).
- Новому сотруднику выделяется персональный наставник из опытных коллег.

Профессиональные требования

- Систематизация профессиональных требований к сотрудникам. Определение и описание перечня текущих и перспективных проф. знаний и навыков.
- Определение целевого уровня знаний и навыков, так для Low-code требуются:
 - Понимание специфики проектирования и работы UI.
 - Принципы построения БД.
 - Навыки тестирования.
 - Понимание бизнес-процессов и их нотаций.
 - Коммуникационные soft skills.

Сопровождение развития

- Составление и исполнение индивидуального плана развития.
- Прохождение курсов на внутрикорпоративных и внешних образовательных платформах.
- Создание внутренних тематических баз знаний.

Выводы

Технические

- ИТ-индустрия демократизируется, растет уровень абстракции разработки
- Эффект от внедрения увеличивается, когда Low\No-code являются «слоями» гибридной платформы совместно с Pro-code
- Low-code платформы подходят для прототипирования, сокращают разрыв между бизнес-пользователями и ИТ для формирования видения будущей системы.
- Автоматизация сложных и высоконагруженных бизнес-процессов требует кастомизации, которой, однако, не стоит увлекаться, чтобы не потерять преимущества Low-code

Связанные с персоналом

- У подхода Low-code много плюсов, но называть его «убийцей традиционной разработки» пока рано, аналитики пока не могут заменить программистов
- Low-code и No-code позволяют работать в графических интерфейсах, но отличаются сложностью и требованиями к квалификации сотрудников
- Методики Low-code и No-code дают возможность компаниям значительно снизить порог входа для сотрудников, желающих принимать участие в разработке программного обеспечения, а соответственно расширить базу кандидатов

Технологии Low Code не заменяют опыт и знания, а дополняют их.

Спасибо за внимание!

Максим Тарасов

Руководитель практики систем
управления бизнес-процессами



НОРНИКЕЛЬ

СПУТНИК