



**МОЖНО ЛИ ИСПОЛЬЗОВАТЬ LOW-CODE
НЕ ТОЛЬКО ДЛЯ СОЗДАНИЯ MVP?**

Тулчинский С.Э.



№1

на рынке
финансирования АПК

№1

на рынке финансирования
сезонных работ

№4

среди инновационных
банков (2022 г.)

№6

по размеру активов



Россельхозбанк — одна из крупнейших и наиболее устойчивых финансовых организаций страны. Универсальный розничный банк для граждан и бизнеса.

Мы определяем вектор развития национальной экономики, поскольку являемся банком № 1 для агропрома.



МИССИЯ

- » Выполняем государственные задачи
- » Внедряем передовые цифровые технологии
- » Поддерживаем доступность 24/7 по всей России
- » для клиентов АПК
- » для малого и среднего бизнеса
- » для населения регионов страны



> 30 тыс.

сотрудников



> 300 тыс.

корпоративных
клиентов



> 10 млн

розничных
клиентов



> 1 400

офисов



> 3 000

банкоматов



82 субъекта

Российской
Федерации

РЕЙТИНГ

AA+ (RU)

Кредитный рейтинг НКР

AA (RU)

Кредитный рейтинг АКРА

A1 (АРРБ)

Наивысший класс

Где подходит no-code или low-code?

LOW-CODE БОЛЕЕ ПРИВЛЕКАТЕЛЬНЫ КОГДА:

- Чем меньше сложность системы
- Чем быстрее нужны хоть какие-то результаты
- Чем более простой результат требуется
- Чем менее ультимативны требования к MVP

ВОЗМОЖНО ПОСЛЕДОВАТЕЛЬНОЕ РАЗВИТИЕ КОДА:

- сначала no-code решение, что-то очень простое
- если оно нравится, но есть ограничения no-code, то выкидывают все и делают в low-code
- если и этого мало, то выкидывают low-code и переходят на фреймворк
- в крайнем случае при ощутимых ограничениях фреймворка — собственное решение с нуля



Другими словами low-code...

1

Подходят не для любой задачи и не для любой стадии жизненного цикла этой задачи

2

Стоят гораздо дороже, чем про них говорят

3

Требуют квалифицированных специалистов

4

Не инструмент разработчика

5

Инструмент бизнеса: быстрый MVP для стартапа

6

Инструмент бизнес-аналитика/руководителя проекта: прототип приложения/ прототип бизнес-процесса

В качестве примеров



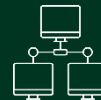
Конструкторы
веб-сайтов



Конструкторы
приложений,
ориентированные
на мобильные
устройства



Конструкторы
приложений,
ориентированные
на интернет



Инструменты
автоматизации
рабочего процесса
(Мидл-варе):

- таблицы, финансы и моделирование
- платежи и транзакции для финансовых рынков
- инструменты для разработчиков мобильных приложений и маркетологов
- аналитика и панели digital маркетинга
- опросы, тесты ...

Почему low-code не позволяет «прыгнуть выше головы»

Нужен рефакторинг кода:
техдолг (в т.ч. дубли
в «картинках») и изменение
архитектуры

1

Автоматизированное
тестирование

2

Микросервисная архитектура
для реиспользования

3

Контроль версий и мердж кода
разных разработчиков

4

CI/CD и другие элементы
инфраструктуры

5

Код внутри low-code
увеличивает требовательность
к описанному выше

6

Сложность обеспечения
требований информационной
безопасности

7

Наличие приводит к росту
«серого» IT

8

Отказ от low-code платформы
будет означать полное
пересоздание ПО

9

Как еще можно использовать

1

Мидл-варе приложения не высокой сложности

2

Собственная платформа

3

Права на разработку и своя команда разработки

4

Гибрид (low-code для визуализации и запуска процессов)

5

Половина на половину (очень правильно структурированный API back-and которые используют фронт-эндеры для сборки сервиса)

Смешанные команды

ГРАЖДАНСКИЕ

- Владелец продукта
- Бизнес-аналитики
- Технологи
- Разработчики и тестировщики пользовательских приложений
- Полупрофильная линия сопровождения
- Частично третья линия

ПРОФЕССИОНАЛЫ

- Архитекторы (корпоративный и др.)
- Скрам-мастер
- Системные аналитики «базовых» систем и интеграции
- Разработчики и тестировщики «базовых» систем и интеграции
- Тестировщики ИФТ, НТ
- Частично третья линия
- Информбезопасность



**БЛАГОДАРЮ
ЗА ВНИМАНИЕ**



TulchinskiySE@rshb.ru