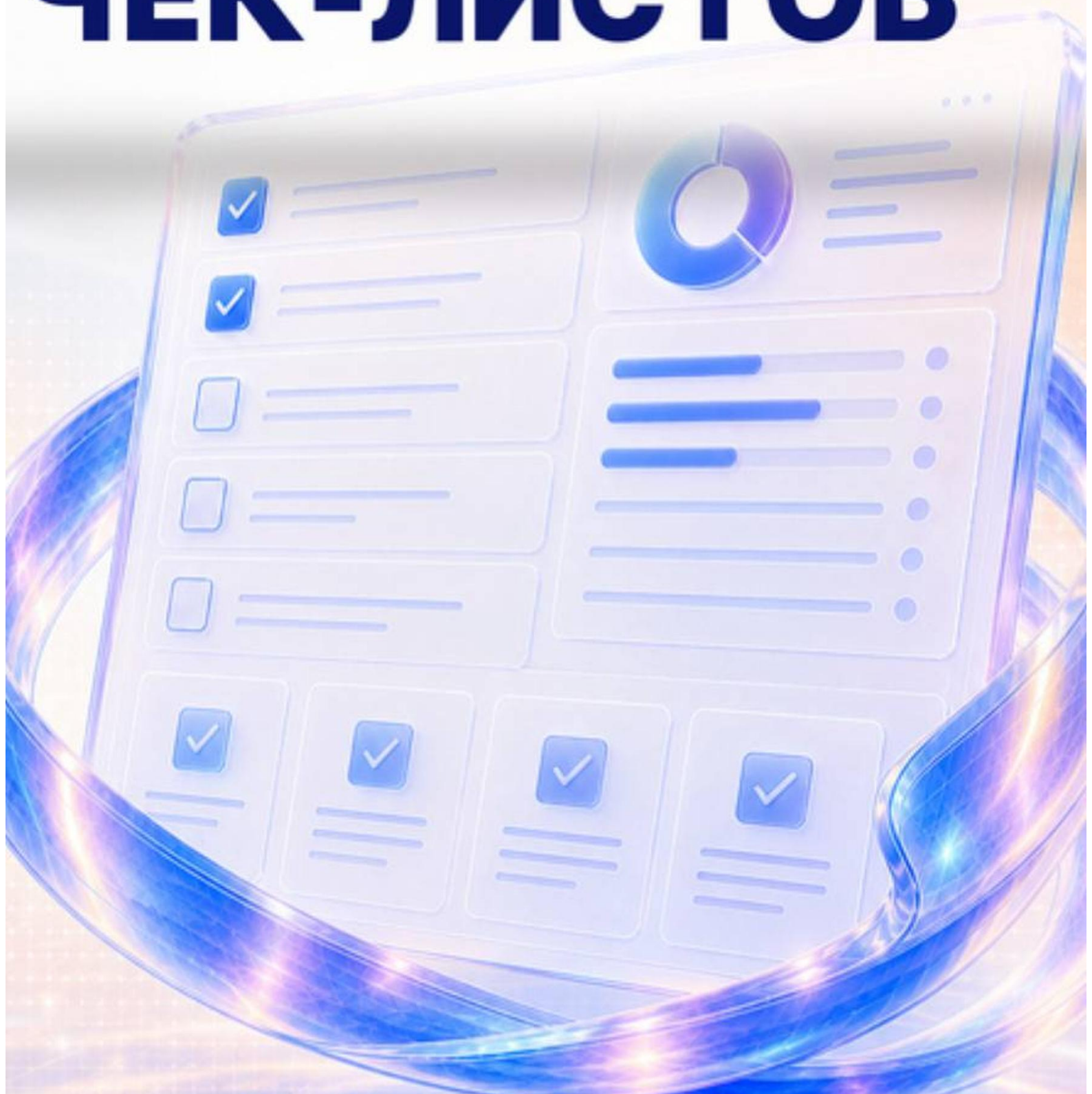


КАК СОЗДАТЬ

КОНСТРУКТОР ЧЕК-ЛИСТОВ



ДИАНА ВАСИЛЬЕВА

Диана Васильева

**Как создать
конструктор чек-листов**

«Автор»

2026

Васильева Д.

Как создать конструктор чек-листов / Д. Васильева — «Автор»,
2026

Повторяющиеся задачи становятся проще, когда каждый шаг виден и ничего не приходится держать в голове. Эта книга поможет создать собственный конструктор чек-листов: с шаблонами, разделами, сроками, отметками выполнения и историей прогресса. Вы настроите создание и копирование списков, сохранение состояния, защиту от случайного удаления и удобный кабинет владельца. В каждой главе есть готовый запрос, ожидаемый результат, ручная проверка и понятный способ исправить ошибку. Руководство рассчитано на новичков и не требует знания программирования. В финале у вас будет практичный инструмент для личных дел, рабочих процессов или клиентских проектов.

© Васильева Д., 2026

© Автор, 2026

Содержание

Как работать с этой книгой	6
Что именно мы создаём	7
Подготовка аккаунта и проекта	8
Техническое задание на человеческом языке	9
Запрос 1. Создаём каркас	10
Запрос 2. Добавляем: создание шаблона по разделам	11
Запрос 3. Добавляем: порядок и обязательность шагов	12
Запрос 4. Добавляем: критерии готовности	13
Конец ознакомительного фрагмента.	14

Диана Васильева

Как создать конструктор чек-листов

Как создать конструктор чек-листов

Создаём в [Chatium](#) инструмент для повторяемых процессов, контроля шагов и понятной приёмки результата

Диана Васильева

Практическое руководство для новичков · 2026

Результат книги: вы создадите веб-приложение, где владелец собирает шаблоны процессов, назначает шаги, добавляет критерии готовности и получает отдельный экземпляр чек-листа для каждой задачи.

Как работать с этой книгой

Эта книга устроена как мастерская. Не нужно сначала читать её целиком. Откройте [Chatium](#), держите книгу рядом и выполняйте шаги по порядку.

В каждом шаге есть пять элементов: действие, готовый запрос, ожидаемый результат, проверка и команда для исправления. Если интерфейс платформы обновился и название кнопки изменилось, ориентируйтесь на смысл действия.

Не пытайтесь получить идеальное приложение одним длинным запросом. Вайбкодинг работает лучше, когда вы создаёте небольшой рабочий слой, проверяете его и только потом добавляете следующий.

Сохраните отдельную копию проекта перед крупными изменениями. Никогда не вставляйте в запросы пароли, данные банковских карт, документы клиентов и другие чувствительные сведения.

Что именно мы создаём

Итоговый продукт — веб-приложение, где владелец собирает шаблоны процессов, назначает шаги, добавляет критерии готовности и получает отдельный экземпляр чек-листа для каждой задачи. Он будет особенно полезен самозанятым, небольшим командам, наставникам, администраторам, организаторам и специалистам с повторяемыми процессами.

У решения будет простой первый экран, понятный сценарий пользователя, проверка введённых данных и сообщение о результате. Мы не будем добавлять функции только ради впечатления.

Ключевой принцип: Готовность процесса = выполненные обязательные шаги / все обязательные шаги; необязательные пункты отображаются отдельно и не скрывают незавершённый критический этап.

Сначала соберём минимальную рабочую версию. Затем улучшим оформление, сохранение данных, мобильный вид, безопасность и способы использования.

Подготовка аккаунта и проекта

Перейдите в [Chatium](#) по ссылке из книги. Создайте аккаунт или войдите в существующий. Один аккаунт платформы можно воспринимать как отдельное пространство: у него собственный адрес, код, таблицы и пользователи.

Создайте отдельный проект для этой книги. Назовите его коротко и понятно, латинскими буквами, без пробелов. Не используйте рабочий проект клиента для экспериментов.

Откройте встроенную среду разработки и помощника. В зависимости от версии интерфейса это может быть веб-редактор, раздел разработки или окно ИИ-помощника.

Перед первым запросом опишите, что вы новичок и хотите получать изменения небольшими шагами. Это снизит риск, что помощник сразу создаст слишком сложную архитектуру.

Техническое задание на человеческом языке

Хорошее техническое задание не обязано быть длинным. Оно должно отвечать на вопросы: кто пользуется инструментом, что вводит, что получает, какие ошибки нельзя допустить и где хранятся данные.

Поля первой версии: Шаблон — название и назначение; Раздел — логическая группа; Шаг — одно проверяемое действие; Критерий готовности — наблюдаемый результат; Обязательность — да или нет; Ответственный — роль или участник; Срок — необязательная дата.

Мы отделяем обязательные поля от необязательных. Для каждого числа задаём допустимый диапазон, для списка — варианты, для даты — ограничения.

Скопируйте запрос ниже в помощник [Chatium](#). Не сокращайте его до одной строки: контекст помогает получить более управляемый результат.

Помоги уточнить техническое задание для инструмента «конструктор чек-листов». Кому он будет полезен: самозанятым, небольшим командам, наставникам, администраторам, организаторам и специалистам с повторяемыми процессами. Результат: веб-приложение, где владелец собирает шаблоны процессов, назначает шаги, добавляет критерии готовности и получает отдельный экземпляр чек-листа для каждой задачи. Поля: Шаблон — название и назначение; Раздел — логическая группа; Шаг — одно проверяемое действие; Критерий готовности — наблюдаемый результат; Обязательность — да или нет; Ответственный — роль или участник; Срок — необязательная дата. Формула или правило: Готовность процесса = выполненные обязательные шаги / все обязательные шаги; необязательные пункты отображаются отдельно и не скрывают незавершённый критический этап. Задай мне не больше семи вопросов. Не предлагай код, пока я не отвечу.

Запрос 1. Создаём каркас

На этом шаге нужен не красивый продукт, а работающий маршрут и экран без ошибок. Просим помощника объяснить структуру файлов и не подключать лишние зависимости.

После выполнения откройте адрес проекта в новой вкладке. Если экран загружается, название видно и кнопка реагирует — каркас готов.

Если вместо приложения появляется ошибка, не просите «исправить всё». Передайте помощнику точный текст ошибки и попросите изменить только необходимый файл.

Я новичок. Создай в [Chatium](#) минимальную рабочую версию инструмента «конструктор чек-листов». Пользователь должен получить следующий результат: веб-приложение, где владелец собирает шаблоны процессов, назначает шаги, добавляет критерии готовности и получает отдельный экземпляр чек-листа для каждой задачи. Сначала перечисли файлы и кратко объясни архитектуру. Затем создай главный маршрут и один экран. Не подключай внешние сервисы и не добавляй функции, которых я не просила. Критерий готовности: страница открывается по ссылке, виден заголовок и основная кнопка.

Проверка: откройте ссылку в отдельной вкладке и убедитесь, что главный экран загружается без технического текста.

Запрос 2. Добавляем: создание шаблона по разделам

Теперь добавим функцию «создание шаблона по разделам». Она должна решать одну понятную задачу и не ломать уже работающие части.

Сначала попросите помощника перечислить файлы, которые он собирается изменить, и кратко объяснить логику. Затем разрешите выполнить изменение.

После обновления проверьте счастливый сценарий, пустые значения и одно заведомо неправильное значение. Не переходите дальше, пока результат не объясним.

Если помощник меняет дизайн, названия полей или формулу без запроса, верните предыдущую рабочую версию и повторите задачу с жёстким ограничением: «не менять существующую логику».

В текущем проекте [Chatium](#) добавь функцию «создание шаблона по разделам». Не меняй существующие названия полей, формулу и рабочие маршруты. До изменения перечисли затрагиваемые файлы. После изменения опиши три ручных теста. Если требуется новое поле данных, сначала объясни его название, тип и обязательность.

Критерий готовности: функция «создание шаблона по разделам» работает в основном и ошибочном сценарии, а предыдущие функции не изменились.

Запрос 3. Добавляем: порядок и обязательность шагов

Теперь добавим функцию «порядок и обязательность шагов». Она должна решать одну понятную задачу и не ломать уже работающие части.

Сначала попросите помощника перечислить файлы, которые он собирается изменить, и кратко объяснить логику. Затем разрешите выполнить изменение.

После обновления проверьте счастливый сценарий, пустые значения и одно заведомо неправильное значение. Не переходите дальше, пока результат не объясним.

Если помощник меняет дизайн, названия полей или формулу без запроса, верните предыдущую рабочую версию и повторите задачу с жёстким ограничением: «не менять существующую логику».

В текущем проекте [Chatium](#) добавь функцию «порядок и обязательность шагов». Не меняй существующие названия полей, формулу и рабочие маршруты. До изменения перечисли затрагиваемые файлы. После изменения опиши три ручных теста. Если требуется новое поле данных, сначала объясни его название, тип и обязательность.

Критерий готовности: функция «порядок и обязательность шагов» работает в основном и ошибочном сценарии, а предыдущие функции не изменились.

Запрос 4. Добавляем: критерии готовности

Теперь добавим функцию «критерии готовности». Она должна решать одну понятную задачу и не ломать уже работающие части.

Сначала попросите помощника перечислить файлы, которые он собирается изменить, и кратко объяснить логику. Затем разрешите выполнить изменение.

После обновления проверьте счастливый сценарий, пустые значения и одно заведомо неправильное значение. Не переходите дальше, пока результат не объясним.

Если помощник меняет дизайн, названия полей или формулу без запроса, верните предыдущую рабочую версию и повторите задачу с жёстким ограничением: «не менять существующую логику».

В текущем проекте [Chatium](#) добавь функцию «критерии готовности». Не меняй существующие названия полей, формулу и рабочие маршруты. До изменения перечисли затрагиваемые файлы. После изменения опиши три ручных теста. Если требуется новое поле данных, сначала объясни его название, тип и обязательность.

Критерий готовности:

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.