

КАК СОЗДАТЬ

УЧЁТ ДОМАШНИХ ЗАПАСОВ



ДИАНА ВАСИЛЬЕВА

Диана Васильева

Как создать учёт домашних запасов

<https://litres.ru/74273002>

SelfPub; 2026

Аннотация

Покупать лишнее, забывать нужное и находить продукты после срока годности — обычная цена домашней неразберихи. Эта книга поможет создать простой учёт запасов с категориями, количеством, сроками годности, минимальным остатком и списком покупок. Вы шаг за шагом настроите добавление товаров, быстрое изменение количества, предупреждения, фильтры и понятную сводку. В каждой главе есть готовый запрос, ожидаемый результат, ручная проверка и способ исправить ошибку. Руководство рассчитано на новичков и не требует навыков программирования. В итоге у вас будет удобный домашний инструмент, который помогает экономить деньги и время.

Содержание

Как работать с этой книгой	5
Что именно мы создаём	6
Подготовка аккаунта и проекта	7
Техническое задание на человеческом языке	8
Запрос 1. Создаём каркас	10
Запрос 2. Добавляем: быстрое добавление запаса	11
Запрос 3. Добавляем: места хранения и категории	13
Запрос 4. Добавляем: расход и пополнение	15
Конец ознакомительного фрагмента.	17

Диана Васильева

Как создать учёт домашних запасов

Как создать учёт домашних запасов

Создаём в [Chatium](#) спокойный учёт продуктов, бытовых средств, сроков годности и покупок без лишних таблиц

Диана Васильева

Практическое руководство для новичков · 2026

Результат книги: вы создадите веб-приложение, которое показывает, что уже есть дома, где это лежит, когда заканчивается срок годности и что действительно нужно купить.

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, которое показывает, что уже есть дома, где это лежит, когда заканчивается срок годности и что действительно нужно купить. Он будет особенно полезен семьям, занятым людям, владельцам дачи и всем, кто хочет покупать меньше дублей и реже выбрасывать забытые запасы.

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: В список покупок попадает только позиция, у которой подтверждённый остаток ниже минимального; одинаковые запасы объединяются лишь при совместимых единицах измерения.

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: Запас — понятное название; Категория — продукты, бытовое или другое; Количество — число и единица; Место хранения — полка, шкаф или морозильник; Срок годности — необязательная дата; Минимальный остаток — порог покупки; Обновление — дата и причина.

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 «учёт домашних запасов». Кому он будет полезен: семьям, занятым людям, владельцам дачи и всем, кто хочет покупать меньше дублей и реже выбрасывать забытые запасы. Результат: веб-приложение, которое показывает, что уже есть дома, где это лежит, когда заканчивается срок годности и что

действительно нужно купить. Поля: Запас — понятное название; Категория — продукты, бытовое или другое; Количество — число и единица; Место хранения — полка, шкаф или морозильник; Срок годности — необязательная дата; Минимальный остаток — порог покупки; Обновление — дата и причина. Формула или правило: В список покупок попадает только позиция, у которой подтверждённый остаток ниже минимального; одинаковые запасы объединяются лишь при совместимых единицах измерения. 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 «учёт домашних запасов». Пользователь должен получить следующий результат: веб-приложение, которое показывает, что уже есть дома, где это лежит, когда заканчивается срок годности и что действительно нужно купить. 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: быстрое добавление запаса

Теперь добавим функцию «быстрое добавление запаса». 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 «быстрое добавление запаса». 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 «быстрое добавление запаса» работает в основном и ошибочном сценарии, а

предыдущие функции не изменились.

Запрос 3. Добавляем: места хранения и категории

Теперь добавим функцию «места хранения и категории». 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 «места хранения и категории». 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 «места хранения и категории» работает в основном и ошибочном сценарии, а

предыдущие функции не изменились.

Запрос 4. Добавляем: расход и пополнение

Теперь добавим функцию «расход и пополнение». 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 «расход и пополнение». 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 «расход и пополнение» работает в основном и ошибочном сценарии, а предыдущие

функции не изменились.

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.