

Софт под задачу: собирают агенты, в работе держим мы.

AI-native сервисная компания. Продаём работающее приложение и его поддержку, а не подписку, часы или код.

 **brimos.dev**

MVP работает от брифа до прода. Можем показать вживую.

Реальный прогон: Telegram-бот от брифа до живого проверенного приложения примерно за 30 минут. Пайплайн прогнан целиком несколько раз на тестовых клиентах.

Journeys / Water Reminder Bot / Water reminder bot with daily streak

Water reminder bot with daily streak

"I need a Telegram bot that reminds me to drink water during the day and tracks my daily streak"

completed took 10 min · 7 requirements · PR #1



Architect's breakdown 4 tasks · 4 done

BACKLOG

0

IN DEV

0

CI

0

DONE

5

✓ Reminder scheduler (every 2 h, 9:00-21:00, user time zone) Done

✓ Glasses, daily goal (/goal) and streak Done

Live log

water-reminder-bot

```
12:27:39 ci ..... [100%]
12:27:44 ci 48 passed in 3.05s
12:27:45 ci CI run 2 passed
12:27:51 brimos Merged pull request #1 into main
12:27:55 deploy Deploy water-reminder-bot
12:27:58 deploy Server: edge-eu-2, 256 MB reserved, port 8031
12:28:03 deploy $ docker build -t water-reminder-bot .
12:28:25 deploy Successfully built water-reminder-bot
12:28:31 deploy Secret TELEGRAM_BOT_TOKEN attached
12:28:37 deploy $ docker run -d --name water-reminder-bot -v bot-data:/app/data ...
12:28:48 deploy Container running
12:28:51 deploy Health: 200 OK · 84 ms
```

Код сейчас получить легко. Работающее приложение никто не берёт.

Конструкторы прототипов

Генерируют код

Выкатить, разместить и чинить владелец должен сам.

Фрилансер

Делает по ТЗ

Поддержка после сдачи — как договорятся.

Пробел

Выкатить, проверить, поддерживать

Владелец без технического опыта обычно не справляется. Эту работу берём мы.

Клиент подтверждает бриф. Все требования записаны до первой строки кода.

Шаг 1 из 4. Одна фраза клиента превращается в семь проверяемых требований.

Journeys / Water Reminder Bot / Water reminder bot with daily streak

Water reminder bot with daily streak

"I need a Telegram bot that reminds me to drink water during the day and tracks my daily streak"

In progress

running for 2 min

7 requirements ^

1

Send a reminder every 2 hours from 9:00 to 21:00 in the user's time zone

2

Log a glass with one tap on the reminder or with /drink

3

Daily goal of 8 glasses by default, changeable with /goal (1-30)

4

Count the streak of days with the goal reached; reset it after a missed day

5

A day ends at the user's local midnight, not at UTC midnight

6

Send a weekly summary on Sunday at 20:00: glasses per day and the streak

7

/stats shows today's glasses, the current streak and the best streak

1

Brief

3 min

2

Plan

1 min

3

Catalog

skipped

4

Code

1 min

5

PR + CI

-

6

Deploy

-

7

QA

-

8

Live

-

PRODUCT OWNER

ARCHITECT

SCAFFOLDER

DEVELOPER

GITHUB CI

DEPLOYER

DEVELOPER

Code

active

started Oct 14, 02:37

finished —

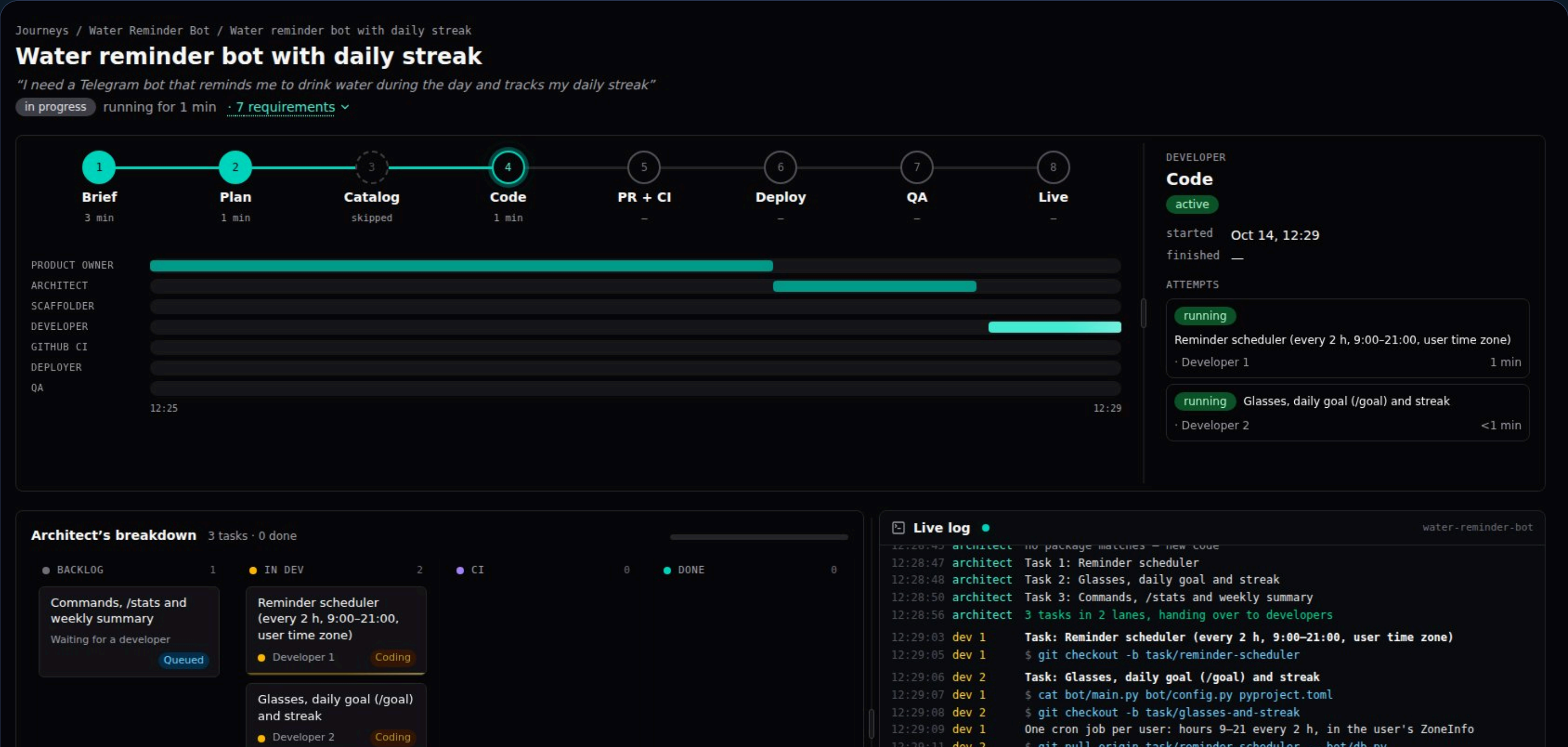
ATTEMPTS

completed

Reminder scheduler (every 2 h, 9:00-21:00, user time zone)

Агент-архитектор делит работу. Агенты-разработчики пишут параллельно.

Шаг 2 из 4. Архитектор сначала проверяет каталог проверенных модулей (спека, тесты, интерфейсы) и общие платформенные сервисы. Новый код пишут, только если ничего не подошло, как в этом демо. Каждый агент работает в изолированном контейнере.



Architect's breakdown 3 tasks · 0 done

● BACKLOG

1

● IN DEV

2

● CI

0

● DONE

0

Commands, /stats and weekly summary

Waiting for a developer

Queued

Reminder scheduler (every 2 h, 9:00–21:00, user time zone)

● Developer 1

Coding

Glasses, daily goal (/goal) and streak

● Developer 2

Coding

Live log

water-reminder-bot

12:28:45 architect no package matches — new code

12:28:47 architect Task 1: Reminder scheduler

12:28:48 architect Task 2: Glasses, daily goal and streak

12:28:50 architect Task 3: Commands, /stats and weekly summary

12:28:56 architect 3 tasks in 2 lanes, handing over to developers

12:29:03 dev 1 Task: Reminder scheduler (every 2 h, 9:00–21:00, user time zone)

12:29:05 dev 1 \$ git checkout -b task/reminder-scheduler

12:29:06 dev 2 Task: Glasses, daily goal (/goal) and streak

12:29:07 dev 1 \$ cat bot/main.py bot/config.py pyproject.toml

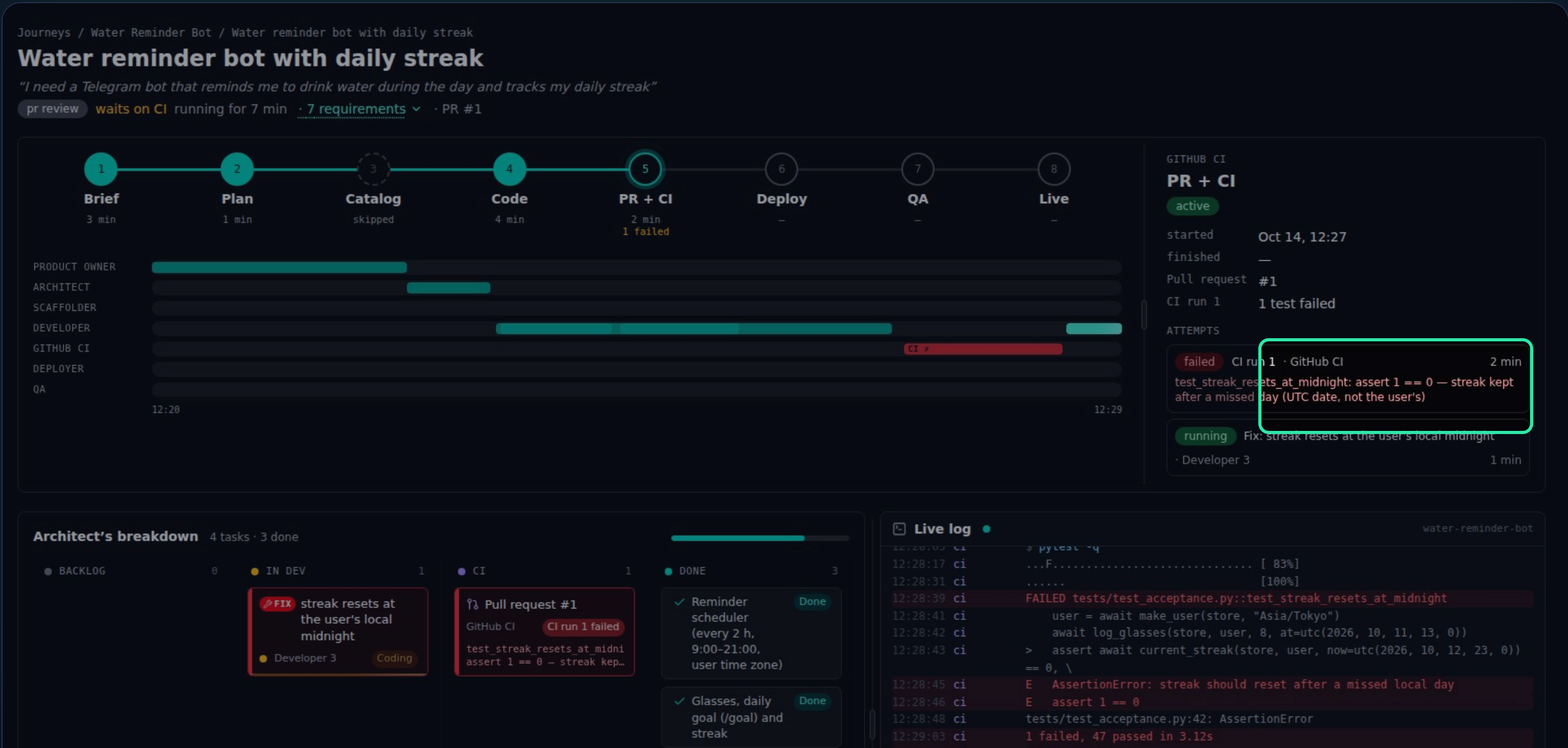
12:29:08 dev 2 \$ git checkout -b task/glasses-and-streak

12:29:09 dev 1 One cron job per user: hours 9–21 every 2 h, in the user's ZoneInfo

12:29:11 dev 2 \$ git pull origin task/reminder-scheduler bot/db.py

CI ловит ошибку. Архитектор сам открывает задачу на исправление.

Шаг 3 из 4. Здесь серия сбрасывалась в полночь по UTC, а не по времени пользователя.
Исправление снова прошло CI до деплоя.



Architect's breakdown

4 tasks · 3 done

BACKLOG

0

IN DEV

1

CI

1

DONE

3

FIX

streak resets at the user's local midnight

Developer 3

Coding

Pull request #1

GitHub CI

CI run 1 failed

test_streak_resets_at_midnight: assert 1 == 0 — streak kept...

Reminder scheduler (every 2 h, 9:00-21:00, user time zone)

Done

Glasses, daily goal (/goal) and streak

Done

Live log

water-reminder-bot

12:28:03 ci

python3 -m pytest -q

12:28:17 ci

...F..... [83%]

12:28:31 ci

..... [100%]

12:28:39 ci

FAILED tests/test_acceptance.py::test_streak_resets_at_midnight

12:28:41 ci

user = await make_user(store, "Asia/Tokyo")

12:28:42 ci

await log_glasses(store, user, 8, at=utc(2026, 10, 11, 13, 0))

12:28:43 ci

> assert await current_streak(store, user, now=utc(2026, 10, 12, 23, 0)) == 0, \

12:28:45 ci

E AssertionError: streak should reset after a missed local day

12:28:46 ci

E assert 1 == 0

12:28:48 ci

tests/test_acceptance.py:42: AssertionError

12:29:03 ci

1 failed, 47 passed in 3.12s

Приложение запускается на наших серверах. Дальше следим за ним мы.

Шаг 4 из 4. Деплой, проверка, что сервис поднялся, QA на живом боте. Все продукты, которые мы ведём, на одной карте. Платформенные сервисы, например чтение Telegram-каналов, запущены один раз и общие для продуктов.

Atlas5 products · 2 placements

Brimosbuilds and deploys

control-1CPU 23% · RAM 9100/16384 MBworkersqueues

Platform servicesrun once, shared by products

tg-reader · 1

Products · grouped by where they run

edge-eu-14 vCPU · 8 GB

Expense Splitteractive

expense-splitter

Habit Trackeractive

habit-tracker

edge-eu-24 vCPU · 8 GB

Morning Tech Digestactive

morning-tech-digesttg-channels

Pet Vaccination Remindersactive

pet-vaccination-reminders

Water Reminder Botactive

water-reminder-bot

Outside worldkeys users gave their products (names only)

TELEGRAM_BOT_TOKEN 5

LayersMatrixProduct

PRODUCT · ACTIVE

Water Reminder Bot

runs onedge-eu-2

containers1

packages0

user keys1

Focus →Latest JourneyProject

Обычная разработка отдаёт код. Brimos отдаёт работающее приложение.

	Фрилансер или агентство	Brimos
Срок до первой версии	Недели: ТЗ, оценка, потом разработка.	Агенты начинают с подтверждённого брифа. В демо живой бот примерно через 30 минут.
Цена	Почасово или за проект. Каждое изменение — новая смета.	Фиксированная цена за результат: за приложение и за каждое изменение, согласуется до работы. Хостинг, мониторинг и исправления по ежемесячной подписке.
Кто держит прод	Клиент или тот, кого он найдёт.	Мы: хостинг, деплой, секреты, мониторинг.
Когда ломается	Владелец замечает и ищет разработчика.	Проверки ловят сбой. Чинит агент, если не вышло, подключается инженер.

Инвесторы ищут AI-компании, которые продают работу, а не инструмент.

“AI-native companies that don’t sell software, they sell the service. Instead of giving you a tool, they just do the work.”

«AI-native компании, которые продают не софт, а сервис. Вместо того чтобы дать вам инструмент, они просто делают работу».

Y Combinator, Requests for Startups, лето 2026 [M9](#)

“For every dollar spent on software, six are spent on services.”

«На каждый доллар, потраченный на софт, приходится шесть на услуги».

Жюльен Бек, Sequoia, в пересказе TEN13 [M3](#)

Агенты уже пишут и чинят код в заданных рамках.

Сложность уходит в хостинг, секреты, мониторинг и поддержку. Этот слой мы и строим.

Крупнейший конструктор приложений идёт туда же.

“Lovable does not output code. The output is a product, and increasingly so, a business.”

«Lovable выдаёт не код. Результат — продукт, и всё чаще бизнес».

Фабиан Хедин, сооснователь Lovable, сентябрь 2026 [M10](#)

Это подтверждает спрос

Компания с оценкой \$13,3 млрд и годовой выручкой больше \$600 млн говорит то же: покупателю нужен работающий продукт, а не код.

Прямой конкурент, рынка хватит обоим

Lovable — прямой конкурент и идёт к тому же: продукт, а не код. Мы рассчитываем, что рынка хватит обоим: мы с первого дня строимся вокруг поддержки приложения в проде, а не вокруг прототипов.

Каждый вариант закрывает часть работы. После запуска ведём только мы.

Вариант	Что получает покупатель	Как считается цена	Кто ведёт после запуска
Фрилансер	Код по ТЗ	Почасово или за проект	Как договорятся
n8n ^{M4}	Платформа для воркфлоу с хостингом	Starter €20 в месяц при оплате за год	Клиент
Lovable и партнёры ^{M5-6}	Хостинг full-stack, проекты делают партнёры	Кредиты и смета партнёра	Клиент или партнёр
Flatlogic, услуги ^{M7}	Заказная разработка под ключ	Загрузка команды на месяц	Команда подрядчика
Brimos	Один процесс, сделанный и поддерживаемый	Фикс. цена за результат и подписка	Наши агенты и инженеры

Модули переиспользуют и другие. Мы продаём ответственность за результат в проде. Каталог модулей из шага 2 пополняется, и переиспользование держит маржу с ростом объёма.

Фиксированная цена за разработку, ежемесячная за то, чтобы всё работало.

Проверим на платных пилотах. Выручки пока нет.

Пилот

Один процесс. Критерии приёмки берутся из подтверждённого брифа. Оплата после приёмки.

Предлагаемый диапазон: \$5–30K и 1–4 недели.

Сопровождение

Хостинг, мониторинг и исправление ошибок по ежемесячной подписке.

Новые функции

У каждого изменения своя фиксированная цена, согласуется до работы. Делаем через тот же бриф.

Ядро команды собрано. Проверенные инженеры готовы подключиться.

В

Владислав Мешкорудный

Основатель, CEO и СТО

С декабря 2025 года строит платформу Brimos.

Около 7 лет в Python-бэкенде и AI-инжиниринге: Яндекс, ЕРАМ, финтех.

github.com/vladmesh · linkedin.com/in/vladmesh

О

Олег Сергиенко

Руководитель разработки

Fullstack-инженер и engineering manager.

Трижды основатель, больше 300 B2B-клиентов.

github.com/okneigres

Ю

Юлия Ким

Операции, с основания

Больше 7 лет ведёт программы и проекты в образовании, сообществах и AI Safety.

Вела операционную часть курса по AI Safety на 150+ участников.

Инженеры, с которыми мы уже работали

Дмитрий Сальников бэкенд

Игорь Тимофеев ML, оценка LLM

Ангелина Колосенок фронтенд

Привлекаем pre-seed на 12 месяцев работы по вехам.

Основатель на полный день, инженеры по вехам, вычислительные ресурсы. Цель к seed:
\$200K ARR или убедительный enterprise-пилот.

**Пройден гейт
готовности**

**Приняты 3–
5 платных
пилотов**

**Измерен
повторный заказ**

**Проверено
сопровождение**

hello@brimos.dev

brimos.dev

Четыре гейта по порядку. Следующий открывается после предыдущего.

Гейт 0. Готовность. До любых ценных данных клиентов: проверенные бэкап и восстановление, отзыв доступа, TLS на эндпоинтах продукта.

Гейт 1. Будут ли платить? 3–5 платных пилотов, каждый принят по своему брифу.

Гейт 2. Окупается ли переиспользование? Второй похожий заказ требует меньше часов инженеров и обходится дешевле.

Гейт 3. Потянем ли поддержку? Пилоты продлевают платное сопровождение, часы поддержки на клиента измерены.

Выход на рынок

Сначала тёплые знакомства через сообщества фаундеров и B2B-сети, 10–20 discovery-звонков до холодного аутрича.

Потом продать пилот с критериями приёмки и оговорённым сроком поддержки по фиксированной цене.

После этого второй похожий заказ на той же основе, с учётом часов инженеров, полной себестоимости и поддержки.

Это план, схема продаж ещё не проверена.

Сколько стоит один заказ и сколько должно сэкономить переиспользование.

Себестоимость заказа

цена – (часы инженеров × полная ставка + вызовы моделей + инфраструктура)

Каждый вызов модели и каждая попытка инженера пишутся в журнал затрат. Пайплайн не привязан к модели, поэтому затраты меряем на тех моделях, на которых работаем для клиентов. R6

Переиспользование и как его измерить

Сейчас: два шаблона приложений, общие runtime-контракты и один пакет в каталоге (напоминания). Инженеры превращают сгенерированные функции в протестированные модули, из них агенты собирают новые приложения. R4–5

По каждому пилоту записываем часы инженеров и полную себестоимость и сравниваем первый заказ с похожим повторным.

2–3 параллельных проекта на инженера — оценка основателя, замеров нет. Цель — 4–6.

Покупатели уже платят за автоматизацию. Входной рынок считаем по пилотам.

Входной рынок, снизу вверх

Достижимые покупатели на первом рынке × годовые расходы на разработку и поддержку. Обе величины берём из discovery-звонков и цен пилотов.

>90% за год

Рост объёма услуг в категории AI Integration and Automation на Upwork, IV квартал 2025. [M2](#)

\$53,5 млрд

Мировой рынок заказной разработки, оценка на 2026 год, прогноз роста 18,8% в год на 2026–2033. Контекст категории, а не наш рынок. [M1](#)

Модули как контракты

Агент собирает продукт из модулей со спецификацией, тестами и проверяемыми интерфейсами, а не из шаблонов.

Прецедент

Crowdbotics собирал приложения из переиспользуемых модулей силами разработчиков и в 2025 году ушёл в анализ legacy-кода. Аутсорсинговые компании продают свои акселераторы. [M8](#)

Что умеет новый продукт сегодня и что нужно до данных пилотов.

Уже работает

- Команды, кнопки и меню в Telegram
- Записи в PostgreSQL сохраняются между обновлениями
- Действия по расписанию, настраиваемые параметры
- Исходящие интеграции на ключах клиента
- Доступ для конкретных пользователей, передача владения

Следующее

- Веб-интерфейс, свой домен, Telegram Mini App
- Входящие вебхуки, платежи, вход через OAuth
- Встроенная почта или хранилище файлов

Работа по гейту 0

- Проверенные бэкап и восстановление данных продукта
- Владелец может отозвать доступ и открыть бота для всех
- TLS на каждом эндпоинте продукта

Публичные источники: рынок, категория и альтернативы.

M1	Grand View Research, custom software development market	оценка на 2026 год; широкая категория, не наш входной рынок
M2	Upwork, Q4 and FY 2025 results	сигнал спроса на автоматизацию
M3	TEN13, AI-native services thesis	цитирует Жюльена Бека (Sequoia) о расходах на услуги
M4	n8n, plans and pricing	тариф Starter, оплата за год; только цена платформы
M5	Lovable Cloud documentation	хостинг приложения, база данных, аутентификация, хранилище
M6	Lovable partner program	партнёры делают и сдают проекты клиентам
M7	Flatlogic, custom development services	разработка под ключ, месячная загрузка, долгая поддержка
M8	Crowdbotics / CoreStory	ребрендинг в 2025 году, теперь анализ legacy-кода
M9	YC Summer 2026 RFS, AI-Native Service Companies	текст раздела RFS
M10	TechCrunch, Lovable crosses \$600M ARR	цитата Хедина и выручка; оценка: TechCrunch, 12 августа 2026

Код, который стоит за утверждениями о продукте, открыт для проверки.

R1	Карта возможностей продукта	что умеют и чего не умеют новые продукты
R2	Пайплайн брифа и доставки	подтверждённые требования, эскалация, релиз
R3	Smoke-проверки после деплоя	HTTP health, проверка бота, состояние контейнеров
R4	README Product Kit	шаблоны приложений, runtime пакетов, ограничения сборки
R5	Каталог пакетов	пакет напоминаний и его версии
R6	Модель журнала инженерных попыток и QA	учёт затрат; неизвестная стоимость так и хранится как неизвестная
R7	Документация по тестам	уровни тестов и пределы того, что они доказывают
R8	README оркестратора	архитектура и список сервисов