

froddy.

# Слой контроля перед выплатой

Один вызов API до каждой выплаты — и система возвращает решение:  
разрешить / на проверку / заблокировать.

PSP · CPA-сети · iGaming · Fintech

[froddy.io](https://froddy.io)

# Ошибки выплат — это не фрод. Это сбои собственной автоматизации

Большинство потерь в автоматизированных выплатах происходит не от внешних атак, а от дублей, сломанных лимитов и ошибок логики. Внутренние контроли их не ловят, потому что часть той же системы.



## Дублирование выплат

Ошибка в логике или гонка процессов запускают одну выплату несколько раз. Обнаруживается позже при сверке.



## Сломанные лимиты

Лимиты настроены неправильно или не срабатывают. Выплаты продолжают уходить, пока проблему не заметят.



## Ошибки выплата-логики

Настройки изменили – контроль сломался. Выплаты продолжают уходить.

*Общее: контроли были внутри той же системы. Когда ломается исполнение — ломаются и контроли.*

ДАННЫЕ

# Публичные оценки потерь из-за ошибочных выплат (2024–2025)

ГЛОБАЛЬНО · В2В АР

## 0,8–2,0%

дублей и ошибочных выплат от годового объёма

Top performers — 0,8%, bottom — 2,0%. Гео: глобальный опрос.

Источник: APQC Open Standards Benchmarking, 2024

ГЛОБАЛЬНО · БАНКИ

## €8 млрд

потерь банков на сбоях транзакций

Сбои исполнения и доставки — самый дорогой операционный риск в банках. Источник: ORX Global Op. Risk Report, 2024

ЕС · БЮДЖЕТ

## 3,6% / €7,1 млрд

неправильных выплат бюджета ЕС

Административные ошибки, не фрод. Снижение с 5,6% (2023).

Источник: European Court of Auditors, Annual Report, 2024

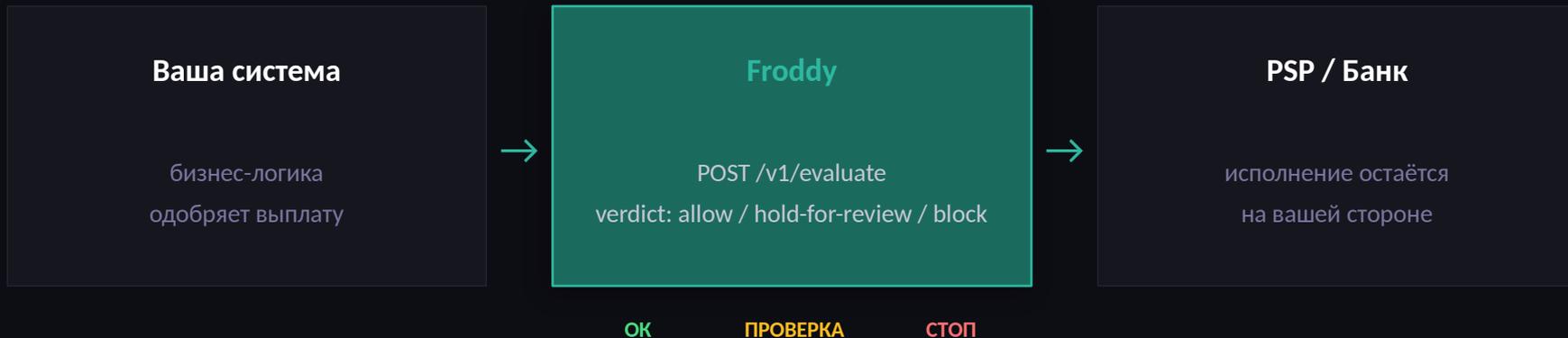
РОССИЯ · БАНКИ

## 20 млрд руб.

ежегодных потерь банков на ошибках выплат

Возвращают только 47%. Двойные списания и процессные ошибки, не фрод. Источник: «Т1 Иннотех», 2024

# Один слой между вашей логикой и движением денег



- Froddy не двигает деньги. Он возвращает решение — что делать с ними, решает ваша система.
- Режим наблюдения (shadow mode): Froddy оценивает и логирует, не вмешиваясь в выплаты. Режим вмешательства включается отдельно.
- По умолчанию — пропускаем. Поведение при недоступности Froddy контролируется на вашей стороне.

# Что конкретно делает Froddy

## ДУБЛИ

### Проверка дублей

Повторный запрос получает то же решение без повторной проверки.

## ЛИМИТЫ

### Лимиты

Лимиты на сумму одной выплаты, общий объём выплат за день и частоту операций за период.

## ПРОВЕРКА

### Hold-механика

Hold-for-review означает: отправить на проверку. Дальше действует ваша система по вашим правилам.

## LOG

### Журнал решений

Каждое решение сохраняется вместе с правилом, данными, итогом и временем проверки.

# Компании собирают это сами — из фрагментов

## Сейчас — без Froddy

- ✗ Инфра выплат провайдера/банка
- ✗ Правила — инхаус или точечное решение
- ✗ Сверка — отдельный вендор или вручную
- ✗ Логи — разрозненные, нет единого журнал решений
- ✗ Изменение правил — ручной процесс



## С Froddy

- ✓ Единый слой правил поверх любого исполнения
- ✓ Проверка на дубли и повторные запросы
- ✓ Лимиты по сумме, объёму выплат за день и частоте операций
- ✓ Журнал решений — все проверки и решения в одном месте
- ✓ Правила меняются через API с сохранением версий

Часто такие предохранители собирают инхаус: это отдельная логика, поддержка и операционная нагрузка. Froddy — тот же класс контролей как внешний слой.

# Кому это нужно

## Платёжные провайдеры / PSP

Автоматизированные расчёты с мерчантами, массовые выплаты.  
Нужен контроль до отправки денег в банк/провайдер.

## CPA-сети / партнёры

Партнёрские выплаты за результат.  
Дубли, аномальная частота и связанные аккаунты — основные риски.

## iGaming

Вывод средств игроков и выплаты аффилиатам. Высокая частота, нестандартные объёмы VIP, бонус-аномалии.

## Финтех / займы / кошельки

Автоматические выплаты и переводы.  
Гонки процессов и повторы запросов — типичные источники дублей.

## Операции / риск / контроль

Командам, которым нужен единый журнал решений до движения денег — без самописного решения.

## Продукт / разработка

Основатели и руководители продукта, которым нужны предохранители для выплат без отдельного сервиса правил.

# Один вызов — один вердикт — один лог

1

## До каждой выплаты

Ваша система отправляет POST /v1/evaluate с идентификатором получателя (entity\_id), суммой (amount) и ключом события (event\_id).

2

## Froddy оценивает запрос

Проверки идут по очереди: сумма выплаты, дневной лимит, частота операций, дубли и повторные запросы.

3

## Вердикт возвращается синхронно

allow → выплата проходит. hold-for-review → отправить на ручную проверку в вашей системе. block → отклонить или отправить на разбор.

4

## Решение логируется

Каждая проверка сохраняется: что проверяли, какие были значения, какое решение приняли и когда.

### POST /v1/evaluate

```
{
  "event_id": "payout_42301",
  "entity_id": "partner_88",
  "amount": 15000.00,
  "currency": "USD",
  "event_type": "payout"
}

// Ответ:
{
  "verdict": "hold-for-review",
  "reason": "daily: $53k / $50k"
}
```

# Почему это полезно бизнесу

01

## Меньше дорогих простых ошибок

Дублированные выплаты и сломанные лимиты — не редкость. Контроль до выплаты ловит их до того, как деньги ушли.

02

## Меньше ручных разборов

Выбранные выплаты – на проверку. Остальное проходит автоматически.

03

## Понятнее правила выплат

Можно точно восстановить, по каким правилам было принято решение.

04

## Журнал аудита

Всегда видно, почему выплата прошла/остановилась и какие правила не сработали.

# Три шага от демо до запуска

1

## Демо

Понять сценарий

Попробуйте интерактивное демо на [froddy.io](https://froddy.io) — запрос / ответ / журнал решений в действии, без интеграции.

[froddy.io/demo](https://froddy.io/demo)

2

## Подключение

Режим наблюдения

Создайте аккаунт и получите API-ключ проекта. POST /v1/evaluate перед вашими выплатами. Режим наблюдения — без автоматического вмешательства в исполнение.

[froddy.io/signup](https://froddy.io/signup)

3

## Режим вмешательства

По готовности

Сначала — наблюдение без блокировок. После настройки лимитов можно отдельно включить режим вмешательства.

Настраивается через API правил

# froddy .

## Посмотрите на продукт в действии

Демо, документация и подключение — на [froddy.io](https://froddy.io)

[Открыть демо](#)

[Начать бесплатно](#)

Почта: [hello@froddy.io](mailto:hello@froddy.io)

Telegram: [@froddysup](https://t.me/froddysup)

Сайт: [froddy.io](https://froddy.io)