

Как мы в ГдеМатериал мониторим заказы

Как мы в ГдеМатериал мониторим заказы

Новые заказы — самый важный ресурс, который у нас есть. За каждый заказ мы заплатили живые деньги: за рекламу, за поисковую оптимизацию, за время холодных продажников.

Ни один заказ не должен потеряться из-за программистов. Есть правило: **видишь проебанный заказ -> бросай все дела и разбирайся**. О каждом инциденте с проебанным заказом нужно написать Арсену, в копию поставив developers@. В письме обязательно указать:

- Причину произошедшего на человеческом языке (не «не проходит валидация по схеме», а «мы обновляли API приёма заказов и не предусмотрели кейс, когда этаж доставки может быть цокольным, т.е. меньше нуля»).
- Что мы сделали (или сделаем) для устранения причины.
- Пользовательские данные, которые были в проебаном заказе: телефон/почта, имя покупателя, содержимое корзины, адрес доставки и т.д.

Как работает мониторинг

Заказ может проебаться на фронтенде или на бекенде.

Бекенд


Проблемы на бекенде мы мониторим в логах. Про любой запрос на `/api/v1/orders/`, на который мы ответили не 201 приходит вот такое письмо:

☆ Papertrail Alerts

17:26 PA

[Papertrail] "non-ok website orders" alert: 1 match (at 2018-06-23 17:26)

To:



1 event matched your [non-ok website orders](#) search in the past hour.

```
Jun 23 17:17:48 website-backend website-backend: [pid: 20593|app: 0|req: 154/4946604] 176.114.214.14 C {48 vars in 1018 bytes} [Sat Jun 23 17:17:48 2018] POST /api/v1/orders/ => generated 99 bytes in 2 msecs (HTTP/2.0 429) 8 headers in 295 bytes (1 switches on core 0)
```

About "non-ok website orders":

- Query: POST /api/v1/orders/ - "HTTP/2.0 201" - "HTTP/1.1 201" - "Broken pipe" program:website-backend
- Run search: [non-ok website orders](#)
- Time zone: Volgograd
- [Edit or unsubscribe](#)

Can we help?
support@papertrailapp.com - help.papertrailapp.com

Каждая строчка означает один проебанный заказ. Если это были вы — напишите об этом. Если не вы — разбирайтесь.

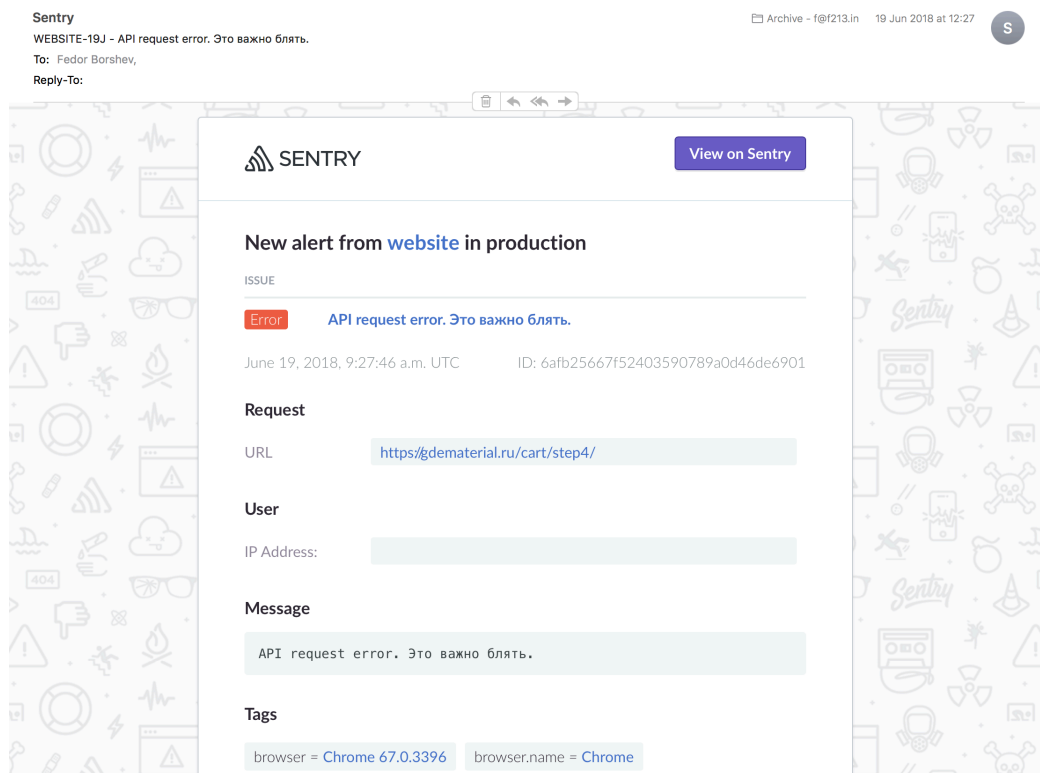
Если код ошибки 4xx, тогда проблема на фронтенде, и нужно смотреть, чем заказ не понравился серверу (см. ниже). Если 5xx или что-другое — проблема в бекенде.

Обратите внимание на код ошибки 429 — он означает «пользователя забанили за слишком частое количество заказов». Это происходит либо потому, что кто-то решил нас заспамить, либо у нас сломался механизм, который разрешает нашим менеджерам делать неограниченное количество заказов.

Анализ проблем бекенда стоит начать с вот [этого](#) или [этого](#) проекта в sentry.

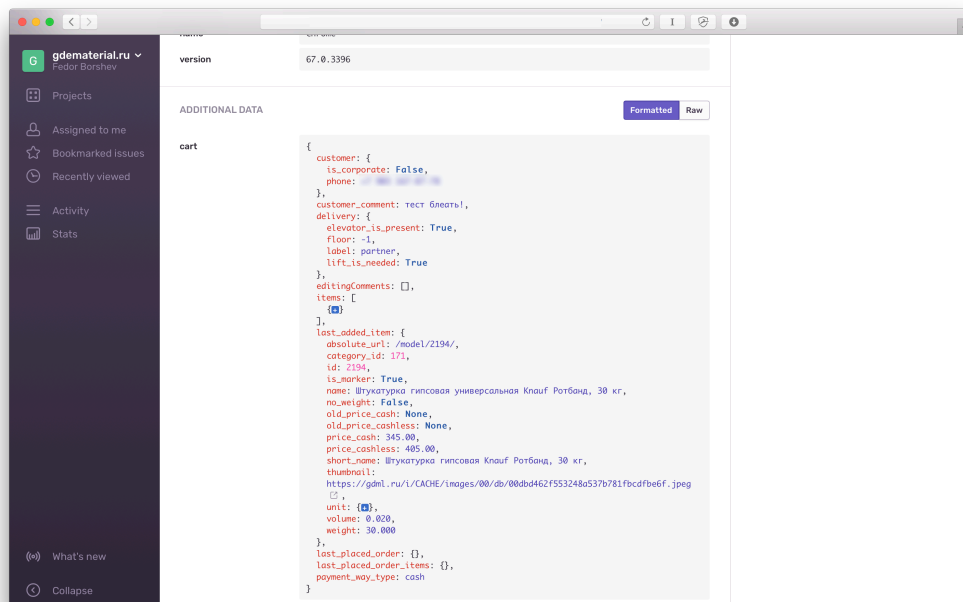
Фронтенд и сентри

Про любой не-GET запрос, завершившийся ошибкой, к нам приходит вот такое письмо от Sentry:



Если при переходе по ссылке вы видите ошибку таймаута или соединения, то скорее всего это проблема у пользователя: если у нас завис бекенд (такого пока ни разу не было), то вы об этом узнаете из мониторинга, а не от sentry.

Основная причина проебанных на фронте заказов — невалидные запросы API. Чтобы вы могли разобрать причину, в ошибке сохраняется ответ сервера и все состояние корзины:



Это же состояние стоит использовать в письме Арсену — он отдаст заказ менеджерам, которые попробуют его спасти.