

Пулл реквесты

- [Философия](#)
- [Оформление ПР](#)
- [Ревью ПР](#)
- [Процесс ревью](#)
 - [Для автора](#)
 - [Для проверяющего](#)
 - [Для обоих](#)
- [Дополнительные материалы](#)

Философия

Ревью ПР помогает:

- повысить качество продукта:
 - выявить проблемы в архитектуре
 - найти баги
 - прийти к единообразию кодовой базы
- развить профессиональные качества:
 - деление знаниями\практиками
 - поиск путей упрощения и улучшения кода

Также, ревью ПР это:

- инструмент для обратной связи
- оценка hard\soft skills разработчика
- источник новых идей для улучшений в процессах, подходах и автоматизации

Оформление ПР

Четко определите назначение ПР в описании

Ответьте на вопросы:

- Какую задачу решает данный ПР?
- Добавляет\удаляет фичу; исправляет баг; рефакторит; покрывает тестами; обновляет зависимость и т.д.?
- Является бейнчмарком или стартом дискуссии?

Передайте контекст

Для качественного ревью – ревьюерам всегда нужен контекст. Имейте в виду, что ПР может читать любой сотрудник компании. Возможно, через месяц\год.

Поэтому кратко, но ёмко опишите *зачем* вообще делалась эта работа (включая релевантные ссылки).

Выделите самое важное для ревью

Выделите в diff ПРа места, для которых вы хотите чтобы ревьюеры уделили *больше* внимания: сложная бизнес-логика, оптимизации рендеринга и т.п.

Не забывайте про WIP\Draft метку, если ПР не готов и вы зовёте ревьюеров на ревью черновика. Т.о. ревьюеры будут уделять внимание только концепции вашего решения и пропускать мимо кодстайл, закомментированный код, логирование и т.д.

Делайте небольшие ПРы

Большой ПР, несущий огромное количество изменений намного сложнее смотреть. Практика показывает, что быстрее и качественнее проходит ревью 3-х пулл реквестов по 400 строк, чем одного на 1200 строк.

В случае, если по тем или иным причинам вам все же надо принести много изменений за раз — разбивайте его на понятные и мелкие коммиты.

В таком случае изменения можно будет смотреть в по-коммитном режиме, что опять же упростит ревью.

Другой вариант: создать родительскую фиче-ветку в которую будут выставляться ПРы с частями большой фичи. Договориться с ревьюерами заранее, что будет несколько ПРов в одну фиче-ветку. Т.о. ревью будет более комфортным и качественным.

Зовите ревьюеров только когда:

- пайплайн прошел успешно
- приложение задеплоилось на тестовый стенд
- в ПР приложена ссылка на стенд

ПР как дискуссия

Не бойтесь открывать WIP\Draft ПРы для обсуждений. Хорошо оформленный черновик:

- удобно донесёт вашу идею
- вовлечёт больше людей, нежели бы вовлёк ваш вопрос в чате
- удобно накапливает идеи для дальнейшего внедрения\рефакторинга

Ревью ПР

Убедитесь в назначении ПР

Какую фичу он привносит; что конкретно исправляет\рефакторит\улучшает).

Иногда, бывает, что автор ПР ошибся в описании, а иногда – в методе решения задачи. Например, автор удаляет модуль, считая, что он больше не используется.

Изучите контекст ПР

Посмотрите описание задачи в Jira, связанные задачи, обсуждения, issues и т.д.

Проверьте выполненность задачи

Следует убедиться, что ПР в полном объеме решает задачу.

На данном шаге также стоит обратить внимание на архитектурные решения, принятые в рамках этого ПР. Возможно, есть более простые и практичные способы решить данную задачу. В таком случае это стоит подробно расписать автору, желательно привести примеры кода и подобной реализации.

Уделите внимание наиболее важным местам

Смотрите указания автора на важные места в ПР. Если таких нет — определите их сами так: бизнес-логика, тесты, чистота кода, комментарии к коду.

А уж в конце по приоритету — название переменных, кодстайл и т.п.

Погрузитесь в реализацию

После проверки выполненности — можно погрузиться в детали реализации задачи: тщательнее смотрите логику модулей, обсуждайте узкие и сложные места. Важным пунктом на этом этапе является консистентность кода относительно всего проекта. Так мы нарабатываем общие практики и подходы.

При обсуждении узких и сложных мест обращаем внимание на наличие документации, это способствует читабельности и поддерживаемости кода в будущем.

Оцените стиль кода

На этом этапе нужно построчно просмотреть код на предмет стилистических ошибок и неявного нейминга.

Это любимая часть плохих ревьюеров и самая малозначимая для проекта и команды. Эта часть должна быть максимально автоматизирована с помощью линтеров\форматтеров.

Цель этого этапа — сделать код максимально понятным и консистентным во всей системе. Не следует пытаться заставить автора сделать все так, как сделали бы вы сами. Читаемость кода — дело субъективное, и тут важны компромиссы. На этом этапе формируется стаилгайд команды, который стоит задокументировать и в дальнейшем оперировать им.

Не спешите с выводами

Не спешите сразу задавать вопросы по блокам кода. Постарайтесь сначала понять всю концепцию ПРа. Бывает, что ответ на заданный вопрос находится чуть ниже по изменениям.

Если ПР вызывает у вас негативную реакцию – возьмите тайм-аут на некоторое время. Не спешите критиковать. Через время — ещё раз все внимательно проведите ревью.

Возможно, у вас просто случилась ошибка коммуникации.

Не затягивайте ревью

Приступая к ревью сразу, вы ускоряете командную разработку и поощряете слать больше атомарных ПР.

Если автор знает, что его ПР будет рассмотрен в ближайшее время и рассмотрение не займет много времени, он будет охотнее разбивать задачи на несколько ПР и их ревью будет гораздо проще и качественнее.

Не делайте лавину замечаний

Часть замечаний, например, можно объединить.



Избегайте

- Вот тут лишний пробел. И тут... И тут



Используйте

- Я заметил, что по всему diff появились лишние пробелы. Давай вернём как было

Не навязывайте доработки, которые не нужны по теме ПРа



Избегайте

- О, ты переименовал модуль... Раздели его типы на публичные и технический, плиз, чтобы не открывать 2 ПР



Используйте

- Кстати, мы всё забываем разделить типы модуля на публичные и. технические. Если у тебе будет время и желание — сделаешь?

Доверяйте

Если в ПР остались только не критичные замечания, то не стоит контролировать их исправление автором и блокировать влитие ПР.

Доверяйте своим коллегам, они точно в состоянии внести минорные правки без вашего повторного контроля. Это ускоряет влитие ПР.

Т.е., если остались только не критичные стилистические замечания-предложения — рекомендуется поставить апрув. Это будет означать, что вы не против того, что автор не исправит эти мелкие замечания и закроет дискуссии.

Просите о декомпозиции ПР

Если вы видите, что в ПР очень много изменений\сложная логика\уточнения порождают новые вопросы — попросите автора ПРа разделить ПР на части (раунды).

Количество раундов ревью может меняться от сложности ПР — нужно стремиться, чтобы их количество не превышало 3-х. Большое количество раундов вызывает напряженность в общении автора и интервьюера.

Отказ от ревью

Бывает, что "рандом" указал на вас, но вы ничего не понимаете в данном подходе\технологии. Не стесняйтесь сказать автору об этом и попросите замену для себя.

Но, при этом, *постарайтесь* таки приложить усилия и понять что происходит в ПР. *Вернитесь* в ПР, когда его посмотрят другие и почитайте дискуссии. Будет полезным опытом.

Процесс ревью

Для автора

Не жалейте делиться информацией

Старайтесь отвечать на каждый комментарий. Старайтесь предоставлять всю уточняющую информацию и рассказывать о причинах тех или иных решений в коде.

М.б. вы упустили что-то в описании ПРа. Поэтому, вам задали вопрос. Отлично! Ответьте автору вопроса и подправьте описание ПРа (внесите в него недостающую информацию).

Не делайте важные правки молча

Если вы согласны с ревьюером по правке — поставьте эмоджи, или напишите текстом "Точно\Согласен\Спасибо".

Если вы просто будете вносить правки автор правок может подумать, что вы:

- его не понимаете
- спешите влить ПР
- не хотите с ним общаться

Правки не всегда нужны

Замечание в ПР может быть: вопросом из любопытства, советом что-то учесть, подсвечивание проблем и (даже) ложным советом. Не стоит сразу переходить к изменению кода по замечаниям.

Стоит понять к какой категории относится замечание.

Уточняйте

Если вам не понятно замечание проверяющего — задайте уточняющие вопросы. Это намного эффективнее, чем додумывать.

Попросите автора дискуссии развернуть свою мысль.

Не игнорируйте дельные замечания

Если вы получили нужное количество апрувов, но вам кто-то подсказал лучшее решение — это не повод игнорировать автора предложения.

Договоритесь внести изменение сразу же после мёрджа данного ПР, чтобы не задерживать релиз фичи, но, при этом, не отложить навсегда дельную доработку.

Не скупитесь на слова благодарности

Если вам подсказали более оптимальное решение, подсветили риски и т.д. — поблагодарите товарища.

Для проверяющего

Объясните, почему необходимо внести правки



Избегайте

- Переделай
- У нас так заведено
- Это опасно



Используйте

- По нашему стайлгайду\мануалу\доке...
- Такого нет в линтере\доке, но мы стараемся так не делать
- Помню из-за такого было падение в проде. Вот ссылка на ПР с фиксом\обсуждение

Конкретизируйте, что именно вам не понятно



Избегайте

- Что это?
- Зачем?



Используйте

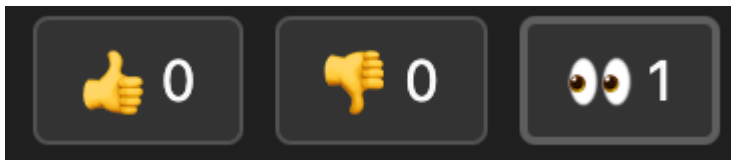
- Стили компонента не должны перетираться сверху
- Почему решил удалить\изменить? Кажется, в этом не было необходимости

Дайте знать, что раунд ревью закончен

Бывает, что во время ревью вы переключились на важную задачу\встречу. Позже, вы возвращаетесь в ПР и заканчиваете ревью.

Новые замечания могут вызвать вопрос в стиле "Почему бы сразу не описать все замечания!?".

Как рекомендация: можно договориться отписываться текстово или ставить реакцию на ПР, которая будет означать, что ревьюер посмотрел все изменения и ждёт лишь правки по своим замечаниям.



Хвалите, если всё хорошо

Не ставьте апрув просто так, если вы полностью согласны с решением\узнали что-то новое благодаря автору. Не поспешите поставить палец вверх.

Для обоих

Будьте явным



Избегайте

- Ты не понял, что нужно было сделать по задаче
- Это работать не будет\Будет ошибка
- Я пробовал другие варианты
- Замени компонент на функциональный



Используйте

- Я не могу найти логику сохранения данных. В задаче говорится именно про сохранение
- Здесь будет ошибка с... [скриншот\текст ошибки]
- Я пробовал другие варианты... [какие именно варианты и какой был результат]
- У компонента нет состояния и не используются методы жизненного цикла. Если перепишем на функциональный компонент, то повысим читаемость и производительность [ссылка на доку]

Предлагайте, а не утверждайте

Это простой и эффективный психологический трюк для облегчения коммуникаций и вовлечение в диалог.



Избегайте

- Давай переделаем на хук
- Давай сразу в хэлпер



Используйте

- Что думаешь насчет ...?
- М.б. сразу вынесем в хэлпер?

Обсуждайте\критикуйте подходы, а не личности



Избегайте

- Ты сделал всё не так
- Твой код не оптимальный



Используйте

- Это не то решение, которое нам нужно
- Кажется, есть более оптимальный вариант

Не используйте негативные эпитеты\гиперболы



Избегайте

- Это **жесть**
- **Никогда** так не делай
- **Всегда** так было
- Так **никто** уже не пишет
- Да мы **бесконечно** рефакторим проект
- **Никому** это не мешало



Используйте

- Не очень оптимальное\надёжное\гибкое решение
- У нас были случаи, когда из-за такого произошло [...]
- Я не помню, чтобы было по другому. Есть примеры?
- Можно сделать по другому
- Этот модуль мы рефакторим уже 3й раз. Кажется, нам нужно переработать его архитектуру
- А глобальная ли это проблема? Давай узнаем у коллег

Не смешивайте контексты в одной дискуссии

Бывает, что в одну ветку начинают вписывать замечания типа "Кстати, я всё хочу в нашем другом проекте использовать этот компонент... А ещё я думаю, что нам нужно обновить линтеры".

Выделяйте для такого отдельные ветки дискуссий\замечаний. Не ленитесь дописывать, что такие дискуссии "на подумать" и не блокируют ПР.

Созвонитесь

Если обсуждение начало перерастать в дебаты — остановитесь и подумайте, имеет ли смысл продолжать диалог в письменной форме.

Практика показывает, что в большинстве случаев гораздо удобнее обсудить все созвонившись, а затем дописать в виде текста краткую выжимку обсуждения.

Будьте молодцами

Будьте скромны, не используйте сарказм\высмеивание\троллинг.

Также, не позволяйте другим так поступать: если видите проявление такого деструктивного поведения — спокойно попросите автора негатива перейти в конструктивное русло. Молчание — не выход.

Дополнительные материалы

<https://habr.com/ru/post/340550/>

<https://habr.com/ru/post/342244/>