

B5 Cover Page

(182x257mm or 516x728pt)

アイのムチ

— レビューで折れないメンタルづくりとレビューアー側の心得 —

[著] ACCESS 技術書典同好会

技術書典 12（2022 年冬）新刊

2022 年 1 月 22 日 ver 1.0

■免責

本書は情報の提供のみを目的としています。

本書の内容を実行・適用・運用したことで何が起こりようとも、それは実行・適用・運用した人自身の責任であり、著者や関係者はいかなる責任も負いません。

■商標

本書に登場するシステム名や製品名は、関係各社の商標または登録商標です。

また本書では、™、®、©などのマークは省略しています。

まえがき / はじめに

ゆるい本です。

目次

まえがき / はじめに	i
第 1 章 よくないレビューの例	1
1.1 審査員形式	1
1.2 試験形式	4
1.3 意識高い系	6
あとがき / おわりに	9

第 1 章

よくないレビューの例

1.1

審査員形式



「実装終わったのでレビューお願いします！」



「アメさん、見てあげれる？」



「ちょっと今忙しくて…。ムチさんに見てもらってください。」



「わかりました！」



「（アメさんがよかったなあ…。ムチさんちょっと怖いんだよな…）」

こういうシーン、みなさんのプロジェクトでも心あたりあるかもしれません。なぜ、ららさんはアメさんにはお願いしやすく、ムチさんにはお願いしにくいのでしょうか。

おや、そう言ってる間にムチさんからレビューコメントが返ってきたようです。

▼ リスト 1.5: @ Muchi commented

NGです。レビュアーやテスターに伝わる説明を書く努力をしてください。
冗長コードも多すぎます。既存の全体設計を事前に見なかったのでしょうか。
ここはif-elseの判定が仕様と逆ですが、わざと壊そうとしていますか？

～翌日～



「（なんか体が重いなぁ。今日は午前休しよう…）」



「あれっ、ららさんは今日どうしたの？」



「午前休って連絡きてましたよ。」



「あ、ホントだ。体が重いって言ってるのが気になるね。心当たりは？」



「昨日、ムチさんのレビューを受けて、遅くまで修正していたようです。」



「なるほど。何かあったのかな。レビュー記録を見てみよう。」



「うーん、厳しい指摘が多いみたいだなあ。スケジュールの心配もあるし、ムチさんに話を聞いてみよう。」

ららさんが午前休を取ってしまったのは、甘えやメンタルが弱いからでしょうか。そう思われては気の毒ですね。ららさんは自分なりによいものを作ろうと一生懸命取り組んでいたはずです。

PMはムチさんをよんでコメントの真意を聞いてみました。



「ムチさん、ららさんの実装ですが、何か大きな問題がありますか？」



「いえ、そこまでは。」



「レビューの指摘が結構重たいようなんですが。」



「あれは、少し勉強不足を感じたので、自分で考えて成長してほしいと期待しました。」

ムチさんの言い分もよくわかります。甘いだけではやっていけないのがプロの世界ですからね。ですが、ららさんはやる気を削がれ、望まない理由で有休まで消費してしまいました。これはららさんと会社の双方にとって、非常にもったいないことですよね。では、このレビューはどうすればよかったのでしょうか。

■審査員形式のレビューとは

- ・複数のレビュアーが辛口審査員のように振る舞う
- ・発表者のコードを攻撃的に批判する

今回は、批評者はムチさんだけですが、ららさんが努力不足だったりよく考えずにやっている、さらに悪意があって間違えたなどと批判するかのようなコメントをしています。

これは発表者のメンタルを壊し、チームには鬱憤とやる気の低下が残るのみなのです。



「（とはいえ、ムチさんは元々そういうスタイルの人で、悪気はないから注意するのみなあ…。そうだ、アメさんにも見てもらおう）」



「アメさーん！ ららさんのコードレビュー、今日はOK ですか？」



「わかりました、今日は大丈夫です。」



「（ムチさんと同じ指摘になるけど、伝え方を変えよう）」

そして、アメさんのレビューコメントがつきました。

▼リスト 1.2: @Ame commented

ご対応ありがとうございます。実装箇所は合ってますが、いくつかお願いしたいことをMust/Wantで伝えますね。

Must: if-elseを仕様通りに直す

Want: 類似実装の共通化、誤りを未然に防ぐための単体テスト作成

説明文はフォーマットを使うとよいと思いますよ。

わからないところは過去のPRを参考にしてみてください。

それを読んで、ららさんは午後からやる気を取り戻したのです。

アメさんのコメントは、審査ではなく、実装者に寄り添うような提案をしていますね。良かった部分は認め、改善点は必須と任意を分けてヒントを出すに留め、実装者自身が考え学ぶ余地も残していますね。



「（さすがアメさん、フォローが上手い！ 普段から心がけているのだろうか…）」

ららさんは1年目で1つ1つの達成を自信に変えている段階なので、それを後退させるフィードバックはチーム力の後退にも繋がります。

気持ちのよいフィードバックとは、必ずしも全員一緒ではないのですが、コミュニケーションを重ねる中で、適切なバランスを見つけていきましょう。

1.2

試験形式



「実装終わったけど今日はアメさん不在だし期限が近いからムチさんに出さなきゃ…。今度こそいいものを出そう！」

▼ リスト 1.5: @ Muchi commented

NGです。このAPIが例外を吐いたときの考慮がされていません。
typoがいくつかあるし、インデントも不揃いです。全体的に不合格。



「ぐすん…コンビニスイーツで元気になろう…。」

ある日の朝会の後、PM はららさんを呼び止め、こう問いかけました。



「ららさんが入って1ヶ月経ちましたね。何か困ってることありますか？」



「私はどうすれば1人前になれるでしょうか。」



「というと？」



「ムチさん曰く私のレビューは不合格だそうで、早く合格して皆さんのお役に立ちたいんです。」



「う〜ん、不合格とは厳しいなあ…。」

PMはアメさんのところに行き、経緯を説明しました。



「ということがあったんですが、アメさんはどう思いますか？」



「ららさんは既に私たちのメンバーですし、私は不合格などと評価したりしないですよ。一度、本来の目的を思い出しましょう。」

■■試験形式のレビューとは

- ・レビューの最後に「合格」「不合格」みたいな用語が飛び交う
- ・タイプミスや改善点を見つけて指摘することを、失敗の通達と捉えている

これは知識が浅い人やニューフェイスがいると起こりがちです。

何かを書いて校正を頼むのは本来試験ではなく、同じ目的を持つ仲間による協力作業です。今回であれば、ららさん、ムチさん、アメさん、PMの4人は、立場こそ違えど「ユーザー満足度の高いソフトウェアにする」などの目的を共有しています。

それが試験形式になってしまうと、発表者はユーザーではなくレビュワーの満足度を高くすることに注力したり、試験に合格することが目的となり、本質からかけ離れてしまいます。また、試験に落ちるという怖れからやる気を損ない、メンバーの協調的な和を乱すことにも繋がります。



「今回は unnecessary な言及を避け、必要な指摘だけをするのが望ましいと思います。私ならこう書きますね。」

▼ リスト 1.5: @ Muchi commented

1. このAPIが例外を起こしたときの考慮
 2. typoやインデントずれ
- これらを直していただければ、残りはLGTMです。



「（この様子だと、アメさんをららさんのメンター係にした方が良さそうだなあ…）」

1.3

意識高い系

それから1ヶ月経ち、ららさんは困ったことは何でもアメさんに相談し、順調にスキルアップしていきました。

そして、ららさんは少し大きな実装を任されました。この間勉強したばかりのアーキテクチャーやデザインパターンを駆使し、わからないところはアメさんに質問しながら、なんとか仕上げました。



「アメさん、色々とうございました！」



「いえいえ。早くマージできるといいですね！」



「（今回はアメさんの言う通りに作ったし、ムチさんも納得してくれるはず…!）」

ららさんはルンルン気分でPRを作り、退社しました。水曜日に早く上がったので、ウェンズデーで見たかった映画を見たり、美味しいものを食べ、オフをエンジョイしました。

翌日、出社するとムチさんの指摘が入っていました。

▼ リスト 1.5: @ Muchi commented

この判定はビジネスロジックなので、サービスクラスでやる仕事じゃないです。もっとドメインを意識して書き直して下さい。

doTask()に2つ以上の責務があり意味不明です。



「(ぐす…アメさんに色々聞いて頑張ったのに…)」



「ららさん、あの件は無事終わりましたか？」



「アメさーん！（泣）」



「ど、どうしたんですか！？」

アメさんは泣くららさんを必死になだめました。
その日の午後、アメさんがPMに呼ばれました。



「ららさんと 1on1 をしたんですが、ちょっと自信を失ってるようですね。」



「はい、私が見て OK でレビューに出してもらったんですが、ムチさんにとっては NG だったようで…。」



「レビューのチェックリストは同じなんですよ？」



「はい。ただ私はそこまで関心の分離を徹底しようと思ってなかったんですが、ムチさんは徹底したかったようです。」



「まあ、その辺のこだわりは、人によって多少は異なりますからね。」



「それに、ららさんは何も考えてないわけじゃなく、ららさんなりの考えでこういう記述にしているなど私は思いました。」



「だから、今はそれで OK を出して、高度な実装は追々学んで欲しかったんですよ。」



「ステップも大事ですからね。」



「ムチさんのフィードバックは、ららさんが頑張って書いた成果物を、突き放すような言い方だったのが気になりました。」

.....

■意識高い系のレビューとは

- ・突き放す感じの指摘で、具体的な修正指示が含まれていない
 - ・タイプミスや改善点を見つけて指摘することを、失敗の通達と捉えている
-

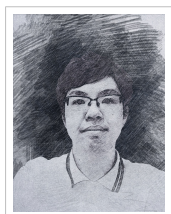
どんな実装にも、レビューーにとってはベストの実装だったり、レビューーなりの考えがあることが多いので、否定だけはよくありません。具体的な修正指示や例を出してあげる方が親切です。

また、答えをそのまま教えればよいとも限りません。あくまで「例」を示し、レビューーが自分で考え、答えはネットや本で見つけるほうが、スキルアップやモチベーションに繋がることがあります。

あとがき / おわりに

いかがだったでしょうか。感想や質問は随時受けつけています。

♣ 著者紹介



Vitantonio Nagauzzi (@tonionagauzzi)

Smartphone App Engineer, "You decide you're happy or not."

- <https://about.me/knagauchi>
- 株式会社 ACCESS

♣ 既刊一覧

- 『ACCESS テックブック』(技術書典 7)

アイのムチ

レビューで折れないメンタルづくりとレビュー側側の心得

2022 年 1 月 22 日 ver 1.0 (技術書典 12)

著 者 ACCESS 技術書典同好会

発行者 tonionagauzzi

連絡先 vitantonionagauzzi@gmail.com

<http://about.me/knagauchi>

@tonionagauzzi (<https://twitter.com/tonionagauzzi>)

印刷所 ねこのしっぽ

© 2022 ACCESS 技術書典同好会