

# 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



「こんにちは、アメです！」



「みなさん、普段レビューしてますか？」



「見る側の方も、見てもらう側の方も、両方の方もいますよね。」



「マリアです。プロジェクトマネージャーをしています。」



「私のチームでは、Git で開発とコードレビューをします。」



「それも長くやってると…色々あるよね、アメさん？」



「そうなんです。」

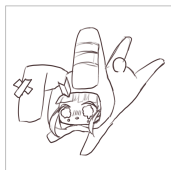


「レビューって、人によって気にする点や出すときの心理が違いますよね。」



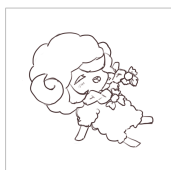
「そんなレビューに楽しく前向きに取り組むため、私たちのチームで取り組んだことを紹介します。」

## 登場人物



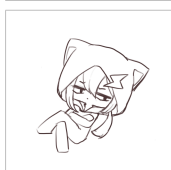
### エマちゃん

- エンジニア 1 年生
- 映画とお笑い甘いものを食べるのが大好き、睡眠は 1 日 9 時間



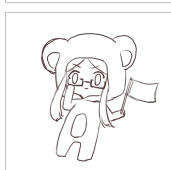
### アメさん

- ゆるふわだけどサバサバなエンジニア（※要相談）
- こだわりは少ないほうで、部屋は物が増えないタイプ



### ムチさん

- 使うものにはとことんこだわりたい寡黙なエンジニア
- 意外とスポーツマン。スキューバダイビングに行く



### マリアさん

- 5 年前にエンジニアから転向したプロジェクトマネージャー
- スプレッドシートより Excel。Slack のバッヂはすぐ消したい派

# 目次

<b>まえがき</b>	<b>i</b>
登場人物 . . . . .	ii
<b>第 1 章    よくないレビューの例</b>	<b>1</b>
1.1    審査員形式 . . . . .	1
1.2    試験形式 . . . . .	5
1.3    意識高い系 . . . . .	8
1.4    持久戦 . . . . .	11
1.5    行き当たりばったり . . . . .	13
1.6    独裁の場 . . . . .	16
<b>あとがき / おわりに</b>	<b>17</b>



---

## 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、エマちゃん  
はアメさんにはお願いしやすくて、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、わざと破壊しようとしていますか？

～翌日～



「（なんか体が重いなぁ。今日は午前休しよう…）」



「あれっ、エマちゃ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。」

エマちゃ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よ。エマちゃ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よ。

マリアさんはムチさんと呼んで、コメントの真意を聞いてみました。



「ムチさん、エマさんの実装ですが、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、エマちゃ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。これはエマちゃ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、エマちゃ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）」



「アメさん、このレビュー指摘だけど、アメさんからもフォローしてくれない？」



「はい。」



「（ふむふむ、なるほど。エマちゃんはこれに悩んでたのか…。私ならこう書くかな）」

### ▼リスト 1.2: @Ame commented

ご対応ありがとうございます。実装箇所は合っていますが、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。



「おはようございます〜。今朝はすみませんでした〜。」



「おはよう。大丈夫？ ムチさんの指摘がちょっと怖かった？」



「あっ、そうじゃないんです。でも、直しててなんか辛くなっちゃいました…。」



「そういうときは、抱え込まず相談してね。私も最初、ムチさんにあんな感じでしごかれたのよね。」



「えっ、そうなんですか！ 意外です。」



「私のほうが後から入ったのもあるけど、あの人、新人だから厳しいんじゃないくて、誰に対してもそうなのよね。」



「でも、ムチさんはきっこう伝えたかったと思うよ。」

それからアメさんは、「私ならこう書く」と考えたコメント通りエマちゃんに伝えました。それを聞いて、エマちゃんはやる気を取り戻しました。

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、実装者自身が考えて学ぶ余地も残しています。

エマちゃんは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…。  
今度こそ何も言われなければいいな！」

## ▼ 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…不合格なんて…。コンビニスイーツで元気になるう…。」

ある日の朝会の後、マリアさんはエマちゃんを呼び止め、こう問いかけました。



「エマちゃんも入って1ヶ月経ったね。何か困ってることある？」



「私はどうすれば1人前になれるでしょうか…。」



「というと？」



「こないだムチさんにレビューで不合格と言われちゃって、早く合格して皆さんのお役に立ちたいんです。」



「う～ん、不合格って言い方は厳しいなあ…。またムチさんに真意を聞いてみようかな。」

ところがムチさんは席にいなかったので、マリアさん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？」



「エマちゃ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。」



「エマちゃんは私からフォローしますね。」



「いつもごめんね、アメさん。」

.....

### ■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が、エマちゃん、ムチさん、アメさん、マリアさんの4人は、立場こそ違えど「ソフトウェアを完成させ、ユーザーを満足させる」という目的は一緒なはずです。

その中で試験のようなレビューをすると、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錯覚します。そして試験に落ちるとやる気を損ない、協調的であるべきメンバーの和が、いつの間にか乱れてしまうのです。



「エマちゃん、それおいしい？」



「あ、これですか！ この前落ち込んだときコンビニで見かけた新作のしらたまあんばんです！ なんかホッとする味です！」



「いいね。私も食べてみようかな。ところで、この前レビューしてた期限の近いあれ、どうなった？」



「時間がなかったのでムチさんに続きを引き取ってもらいました…。」



「えーん、思い出したら悲しくて、もう 1 個食べなくなっちゃいました〜。買ってきます〜。」



「待って待って！ あれね、実は私も途中まで見てコメントしかけてたの。これ見て。」

#### ▼ 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。



「（LGTM…そうだったんだ。よかった…。間違えたところは、次は気をつけよう）」

それから、エマちゃんのしらたまあんぱんブームも終わり、気になってた体重もそれほど増えずに済んだのでした。

アメさんがエマちゃんに見せたコメントは、評価などの不必要な言及を避け、そのとき必要な指摘だけを伝えるものでした。そして、やはり良かった部分は認めています。

何かを書いて校正を頼むのは試験ではなく、同じ目的を持つ仲間同士の協力作業なんです。

## 1.3

### 意識高い系

それから1ヶ月経ち、エマちゃ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きました。

そして、エマちゃ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、色々ありがとうございました！」



「いえいえ。早くムチさんからも Approve をもらえるといいね！」



「（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…!）」

エマちゃんはルンルン気分で PR を作り、退社しました。水曜日だし、ウェンズデーを活用して前から見たかった映画を見たり、ケーキバイキングに行ったりして、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来ていました。

#### ▼ 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書き直しだなんて…）」



「エマちゃん、Approve もらえた？」



「アメさーん！（泣）」



「ど、どうしたの!？」

アメさんは泣くエマちゃんを必死になだめました。その日の午後、アメさんがマリアさんに呼ばれました。



「エマちゃんと 1on1 をしたんですが、ちょっと自信を失ってるようだね。」



「はい、私がレビューして OK だったんですが、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ポイントは 2 人とも同じなんだよね? なんで分かれたの? 」



「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。」



「それに、エマちゃんは何も考えてないわけじゃなく、エマちゃんなりの考えでこういう記述してるんだなって感じて、尊重してあげたかったんです。」



「だから、今は OK を出して、高度な実装は追々学んでくれるといいな~と思ったのが、私側の感覚ですかね。」



「なるほどね。アメさんから見て、ムチさんのやり方はどう思う? 」



「厳しめに見てくれるのはありがたいんですが、頑張ってるのに突き放すような言い方はよくないかなって思いますね。」

#### ■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、レビューイヤーにとってはベストの実装だったり、レビューイヤ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、レビューイが自分で考え、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。

（ここで4コマ漫画を挿入したい。ストーリーは、マリアさんがみんなをランチに誘ったらムチさんが「大丈夫です」と断る。「大丈夫ですってなにー！？」ってツッコミが入る）

## 1.4 持久戦

さて、レビューを受けるのはエマちゃんだけではありません。

マリアさんとアメさんは、今後のプロジェクト計画についてプロジェクト責任者やステークホルダーたちと対面でのレビューを受けました。しかし、その会議は予定終了時刻を2時間もオーバーした上、いくつかの議論は行われませんでした。



「遅くまでお疲れ様。」



「あの方々とやると、いつも予定時間を超えますよね〜。」



「ホントにね、やんなってくる。」



「それに最初は段取り良かったけど、途中から同じ話の繰り返しでした。」



「終わりのほうは、向こうも疲れてたしね。」



「しかも結論がいくつか出ないと思うので、別の日に再レビュー設定することになりそうです。」



「最初の1時間で切り上げてそんなに変わらなかったよね。」



「決めた予定だから最後までやりたかったんでしょうね。」



「今頃向こうは、やった！ って満足感だけ残ってるんじゃない？」

実はこのレビュー、向こうも満足感ではなく、マリアさんと同じようなことを感じ、愚痴っていたのです。

つまり、誰か特定の人の振る舞いが良くなかったわけではなく、お互いにこのまま続けると非効率とわかっていながら、引っ込みがつかなかったのです。

.....

■持久戦のレビューとは

- ・やたらと長く、予定を超過する
  - ・最初はやる気があっても、時間が経つと疲れる
  - ・発表者もレビュアーも、時間が経つにつれ不注意になるが、切り上げられない
  - ・体力を吸われ、レビューの精度も低くなる
  - ・充実感を伴って終わり、本質的な問題に気づかない
- .....

とはいえ、対面での議論を途中で切り上げるのは、思い切りの要ることです。スケジュール調整も必要なので容易ではありません。

最初に「この場ではこれだけ話す!」「議題が逸れたらチャットツールやチケットなど、非同期な方法で続きを議論する」など、あらかじめ合意しておくことが重要でしょう。

## 1.5 行き当たりばったり

エマちゃんたちの会社は、360 度評価が採用されています。今日はムチさんがメンバーからの評価フィードバックを受ける日です。



「ムチさんのプロジェクト貢献度、コードの質の良さ、ドキュメントの更新頻度はおおむね高い点数でした。」



「はい。」



「良い点として、ベースの知識の豊富や、解決力、実装の早さなどが挙げられました。」



「はい。」



「一方、改善点として、円滑なコミュニケーションがややしづらいことや、他の方との交流が少なく、話しかけづらいなどがありました。」



「なるほど。」



「改善点について、思うことはありますか？」



「特に…ないです。」



「（ムチさん、毎回こうなんだよな～。本心が読めない…）」

席に戻ったムチさんは、さっさと帰り支度を始めました。

実はムチさん、帰りに書店でインテリア雑誌を立ち読みし、良い本なら買うのが日課です。今日も観葉植物を替えたくて新刊を読んでいます、どうしても落ち着きません。360 度評価をきっかけに、過去数ヶ月のことが気になり始めたようです。



「(今まで通りやってるはずなのに、新人が配属されてからなんかペースが狂うなあ…)」



「(アメさんが人によって態度やレビューの許可基準を変えるから、僕がうまくフォローしてるつもりなんだけど)」



「(このままレビュー品質が担保されなくなる前に、PMに警告しようかな)」

翌日、ムチさんは Slack でマリアさんに「プロジェクトに潜在リスクがあります」と一言送りました。何のことも見当つかなかったマリアさんは急いで戻り、ムチさんと直接話すことにしました。



「…じゃ、リスクはバグとかのことじゃないんだね。」



「ええ。アメさんをレビューアールとして教育すべきだと思います。」



「(ムチさんが、気になることを自分から話してくれたのは初めてだなあ)」



「(でも、エマちゃんのフォローとか頑張ってくれてるアメさんに、何か足りない点があるんだろうか…)」



「アメさんのどのあたりが足りてないの？」



「レビューが行き当たりばったりなところかと。」



「行き当たりばったり…か。私にはよくわかんないけど事実としてムチさんはそう感じてるってことだね？」



「はい。」



「（もしかして、この前の OK と NG が 2 人の間で分かれた話が何か関係してるのかなあ）」



「わかった、アメさんにも意見聞いてみるね。」



「よろしくお願いします。」

マリアさんは、後で行き当たりばったりとは何かを何度も考え直しました。一般的には、次のような状態を指します。

#### ..... 行き当たりばったりのレビューとは

- ・複数のレビュアーがあり、人それぞれレビューポイントや評価基準が大きく違う
  - ・同じレビュアーでも一貫性がない。例えば、グローバル変数のある日は禁止し、別の日は許可する
  - ・前と同じ問題が見つかって、前と同じ指摘が行われない。言うのを躊躇う。同じ指摘は 2 度と行われない
- .....

これらは人数が増えれば起こりがちです。ここで間違えないでほしいのが、「評価基準が"大きく"違う」と書いたように、多少は違ってよいのです。人間の目によるチェック漏れは 0 にできないし、レビュアーのスキルや経験に応じてレビューポイントが違うのも当然です。例えば可読性の低さやブラックボックス化の懸念など、不慣れな人のほうが気づきやすいこともありますね。

ですが、あまりにも行き当たりばったりな場合、発表者が気づいて理不尽さを感じたり、鬱憤、やる気の低下に繋がりがかねません。

また、チェック漏れによる品質低下リスクも高まります。行きすぎないよう、Try と改善が必要です。

## 1.6

### 独裁の場

- リモートワークになり、みんなのコミュニケーションが希薄になったところでトラブルが発生し、緊急のコードレビューでバタバタする回

残り

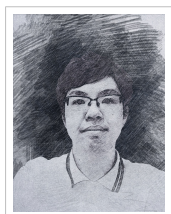
レビューア側の気をつけること\* 独裁の場\* 論点がズレている\* 一方通行

レビューイ側の気をつけること\*

# 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ア側的心得

---

2022 年 1 月 22 日 ver 1.0 (技術書典 12)

著 者 ACCESS 技術書典同好会  
イラスト (好きな表記を教えてください to Umemoto さん)  
発行者 tonionagauzzi  
連絡先 vitantonionagauzzi@gmail.com  
<http://about.me/knagauchi>  
@tonionagauzzi (<https://twitter.com/tonionagauzzi>)

---

© 2022 ACCESS 技術書典同好会