A Appendix

I.A Principal Component Analysis

To identify common themes in the data, we use principal component analysis (PCA). We first transform each comment into a vector of words. We then strip the comments of "stop words," such as "the", "a", "we", and "she" and use stemming to join together words like "test," "tests," and "testing." We then limit to words that appear in at least one hundred distinct comments. We then count the number of times each of these words appears in each comment. Even this more parsimonious representation of the code-review data is high-dimensional, with over a thousand variables. To interpret the data, we reduce its dimensionality with PCA. PCA transforms the data into a new coordinate system, in which most of the variation can be described in fewer dimensions. Table A.1 shows illustrative examples of the primary principal components. Figure A.2 show how these components differ across one- and multi-building teams before and after the offices closed.

I.B Crowdsourced Comment Evaluation

We asked external evaluators to rate the quality of a random subset of comments along several dimensions. We recruited the evaluators through Upwork, selecting workers whom Upwork flagged as being top in the programming languages used by the firm. All the evaluators worked as software engineers, knew the programming languages used by the firm (e.g., PHP or Java), and had both written and received code reviews. For each comment, the engineers were asked to imagine that they had received the comment on a piece of code that they had written. They were then asked to respond to the following questions:

- Would you find this comment helpful?
- Do you think you would change your code because of this comment?
- Does this comment suggest actionable steps to change your code?
- Does this comment explain the reason for changing your code?
- Is the tone of this comment rude?

For the first four questions, the crowd-sourced engineers could answer "yes," "no," or "not enough information." For the question about tone, they could answer "No," "A little bit," "Moderately," "Very," or "Not enough information."

A total of 5,377 comments were evaluated by 22 software engineers. Comments were selected at random, stratifying by pre-post period, one- versus multi-building teams, and engineer gender. Each comment was stripped of any firm-specific content (e.g., the name of the firm) or code that may contain sensitive information. Table A.2 shows a random subsample of comments along with their evaluations.

For any particular dimension, engineers said they did not have enough information to rate between 4 to 26 percent of the comments. Of the comments that could be evaluated without additional information, 87 percent were considered helpful, 68 percent were deemed to be actionable, 70 percent were seen as likely to result in changing code, 58 percent gave a reasoning for the change, and 85 percent were considered to not be even a little bit rude.

The crowdsourced evaluations were provided by experienced engineers. Sixty-eight percent worked as software engineers for 5 or more years. All of them had some college and 86 percent had a college degree. These engineers had all written and received code reviews in the past, having received approximately 600 reviews and written approximately 560 reviews on average. Additionally, to verify that the engineers were sufficiently competent to provide meaningful evaluations of the comments, we conditioned their participation upon successfully answering the following technical questions.

- What is the time complexity of the following Python function that finds the maximum element in a list?

```
def find_max_element(lst):
    max_element = lst[0]
    for element in lst:
        if element > max_element:
            max_element = element

    return max_element
```

  - O(1)
  - O(n)
  - O(log n)
  - O(n^2)

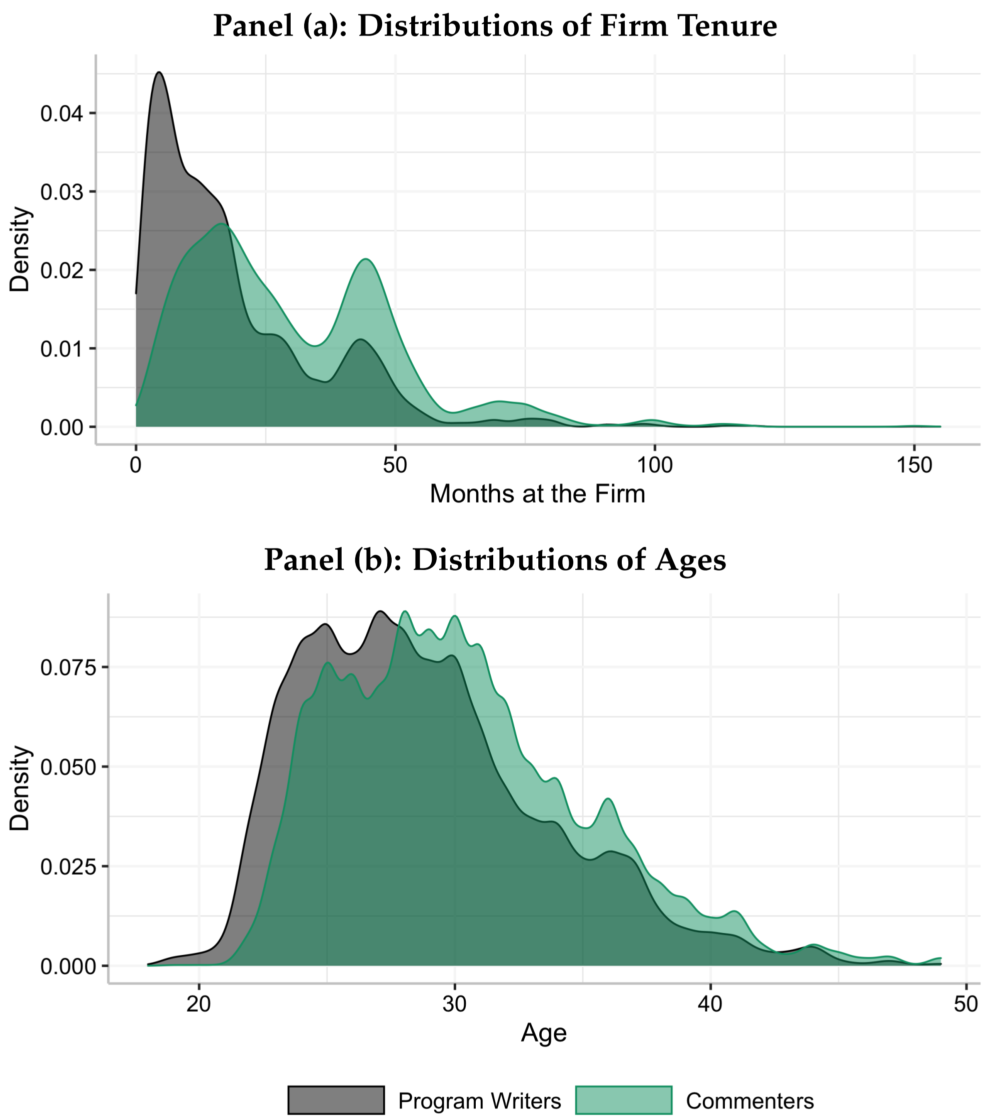
- Suppose you have an array of integers in ascending order. You need to find a target element in the array and return its index. If the target element is not present in the array, you should return -1. Which of the following algorithms would be most appropriate for this task?
  - Linear Search
  - Binary Search
  - Depth-First Search (DFS)
  - Breadth-First Search (BFS)

- Which of the following data structures is typically used to implement a Last-In-First-Out (LIFO) behavior?
  - Linked-List
  - Queue
  - Hash Table
  - Stack

We included five overlapping comments to calculate measures of inter-rater reliability. The engineers' responses adhered with the median response more than seventy percent of the time for each of the rated dimensions. The engineers have Fleiss' Kappa measure of inter-rater reliability of $\kappa = 0.09$ for comment helpfulness, $\kappa = 0.23$ for actionability, $\kappa = 0.28$ for implementability, $\kappa = 0.16$ for including rationale, and $\kappa \approx 0$ for the tone of the comment, which they deemed to be almost never rude.

I.C Figures & Tables

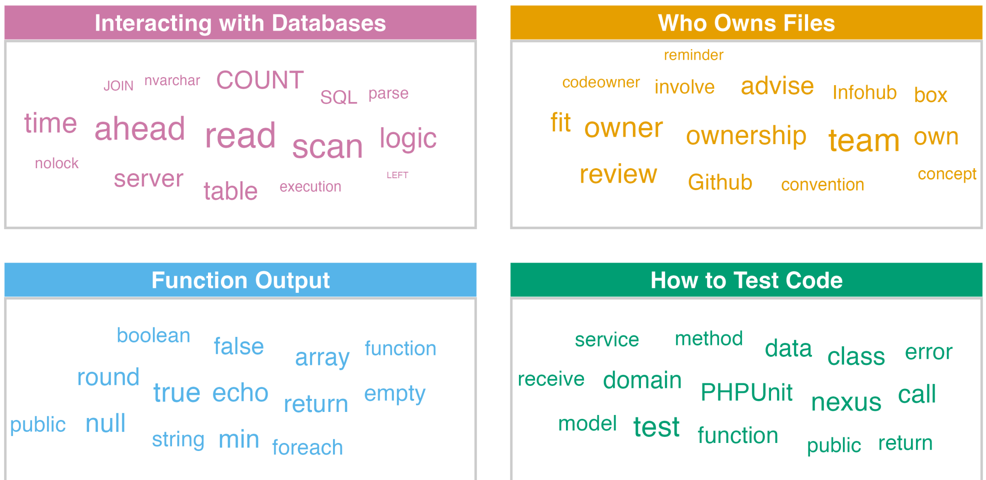
Figure A.1: Program Writer and Commenter Age and Experience



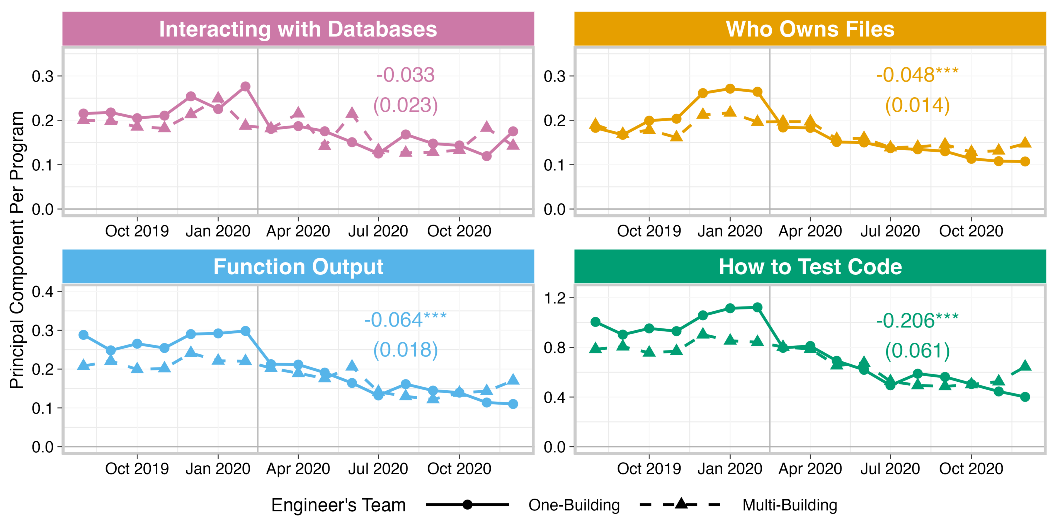
Notes: These figures contrast the experience of program writers and comments (a) at the firm and (b) in their careers. The grey histogram shows the densities for engineers who write programs, weighted by the number of programs that they write. The green distributions show the densities for engineers who write comments on code, again weighted by the number of programs they comment upon.

Figure A.2: Themes in Programmers' Online Peer Feedback

Panel (a): Word Loadings in Principal Components

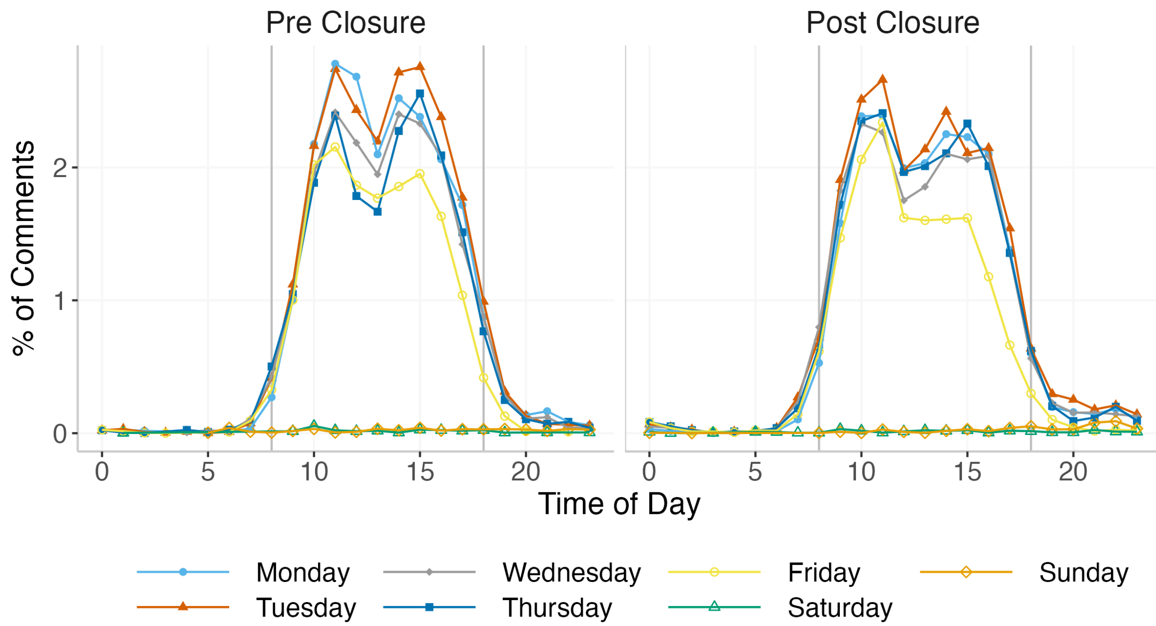


Panel (b): Proximity and Themes in Feedback



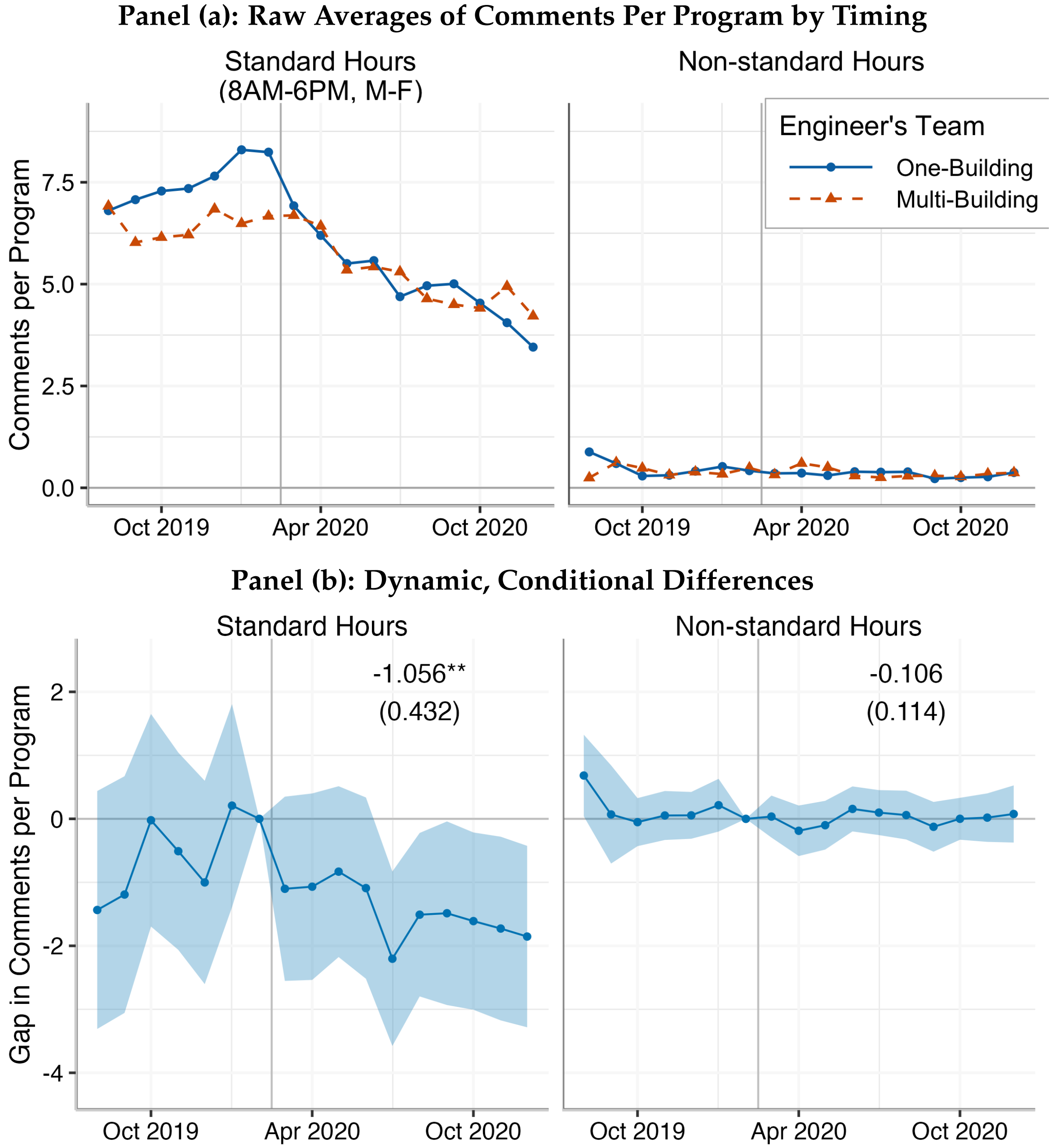
Notes: This figure illustrates the common themes in programmers' online feedback to one another, and how proximity impacts these themes. These themes are identified using principal component analysis (see Section I.A for details and Table A.1 for emblematic example comments). Panel (a) presents the words with the highest loading for each component, with the size reflecting the loading. The first component identifies comments that are about how to *read* data from databases often in the structured query language, *SQL*. The second component identifies two, typically non-overlapping groups of comments. One is about asking *owners* of code on *Github* to *review* suggested changes. The other is about what *functions return*, with special attention to edge cases, like *null* values and *empty arrays*. The final component shown here is about *testing* code, often using the testing suite *PHPUnit*. Panel (b) replicates the analysis in Figure 1 for these components in the comments on each program. The annotated coefficient is the difference-in-differences estimate conditional on our preferred controls. Standard errors are clustered by engineering team. *p<0.1; **p<0.05; ***p<0.01.

Figure A.3: Timing of Comments Over the Course of the Day



Notes: This figure plots the timing of comments over the course of the day. The x-axis plots the time of day in military time. The y-axis is the percent of comments that occur in that particular time of day on that particular day of week. The left plot is the period before the office closures of COVID-19. The right plot is the period after the office closures of COVID-19. The vertical lines highlight typical office hours from 8am to 6pm.

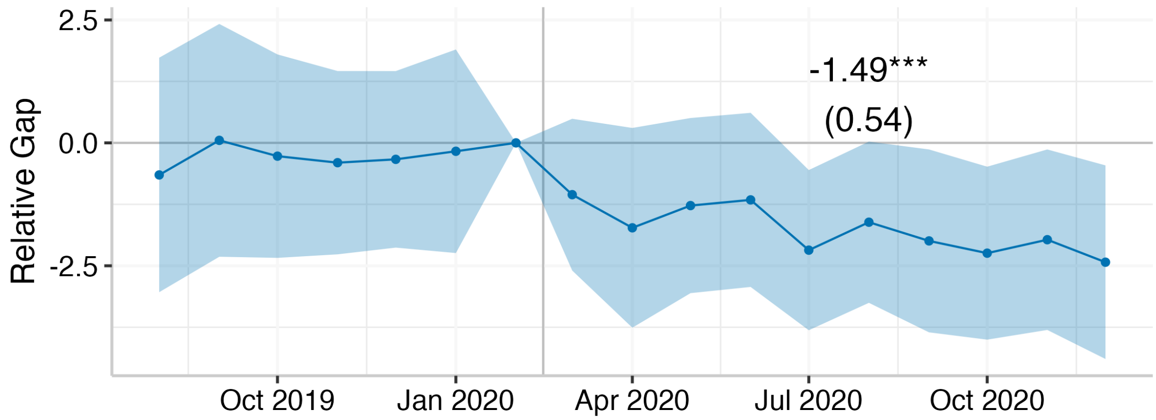
Figure A.4: Proximity to Teammates and Online Feedback Inside and Outside of Standard Work Hours



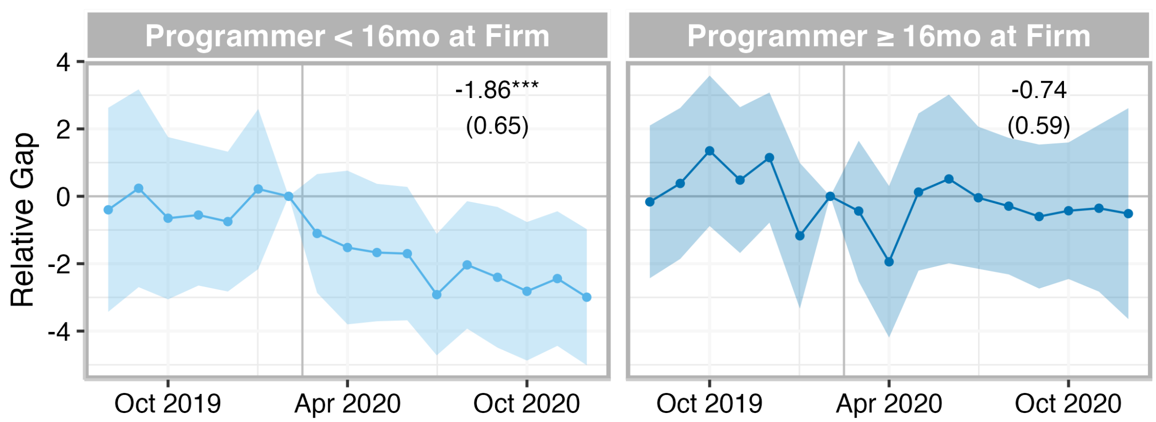
Notes: This figure illustrates the online feedback received by engineers in one-building teams (N=637) and engineers on multi-building teams (N=418) before and after the offices closed for COVID-19 (the grey vertical lines). The left plots consider comments given in standard work hours (8AM to 6PM, Monday through Friday); the right plots consider comments given in other times. Panel (a) plots the raw averages, while Panel (b) plots the differences, conditional on our preferred controls for program scope, engineering type, and tenure. The ribbon is a 95% confidence interval with clustering by engineering team. The annotated coefficient is the difference-in-differences estimate from Equation 2. Only engineers whose teammates all worked in in the main campus are included. *p<0.1; **p<0.05; ***p<0.01.

Figure A.5: Proximity to Teammates' Impact, Conditional on Engineer Tenure by Age

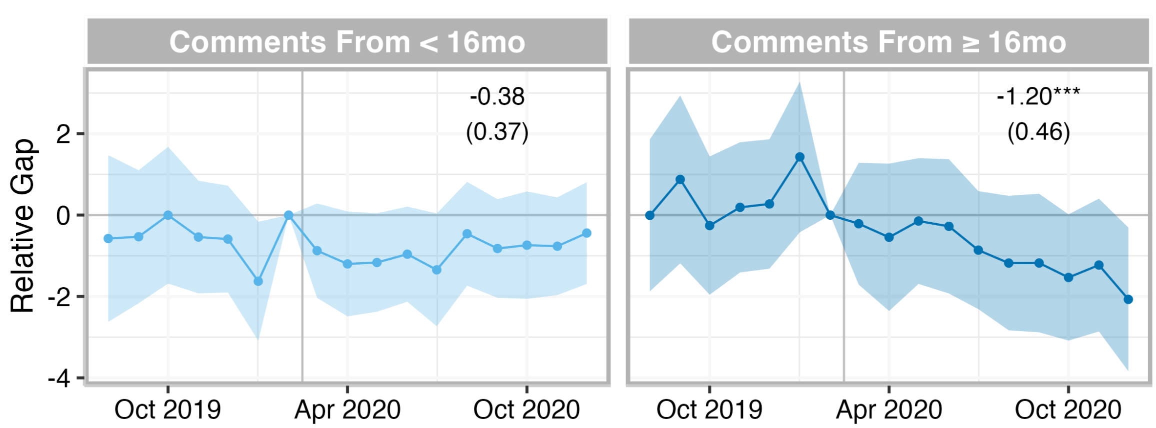
Panel (a): Dynamic Differences in Comments per Program



Panel (b): Dynamic Differences in Comments Per Program by Program Writer's Tenure

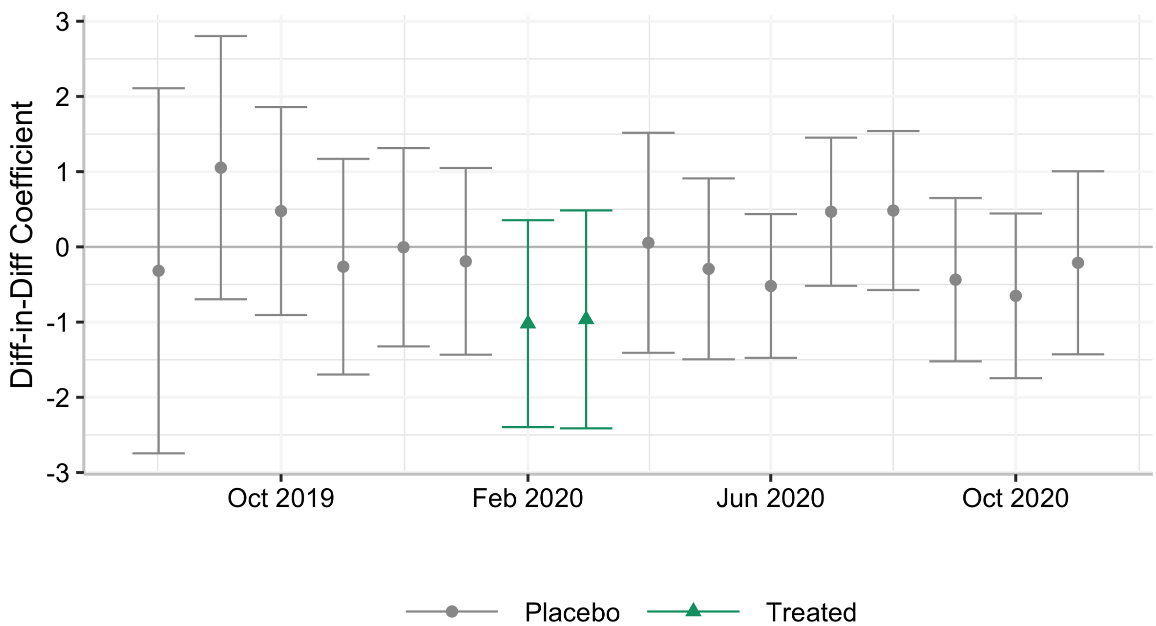


Panel (c): Dynamic Differences in Comments per Program by Commenter's Tenure



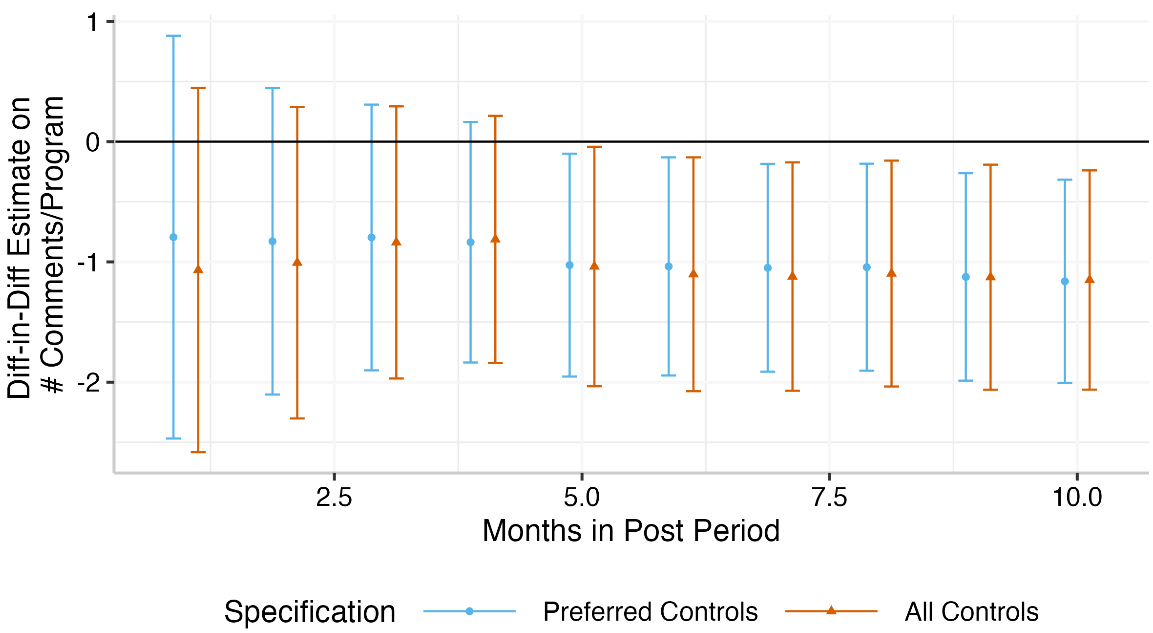
Notes: Panel (a) replicates the analyses in Figure 1(b), including additional time-varying controls for tenure and age. This allows us to remove the effect of a hiring spell that occurred in the pre-period. Panels (b) and (c) replicate Figure 2 with the same set of controls.

Figure A.6: Placebo Treatment Dates' Effects of Proximity on Online Feedback from Coworkers



Notes: This figure illustrates difference-in-differences estimates that compare the change in comments for engineers on one- and multi-building teams in two-month bandwidths. The grey circles show periods that do not include the treated window; the green triangles include the treated window. All regressions include our preferred controls for engineering type, engineer tenure, and program scope (in column four of Table 2). The error bars are 95% confidence intervals with standard errors clustered by engineering team.

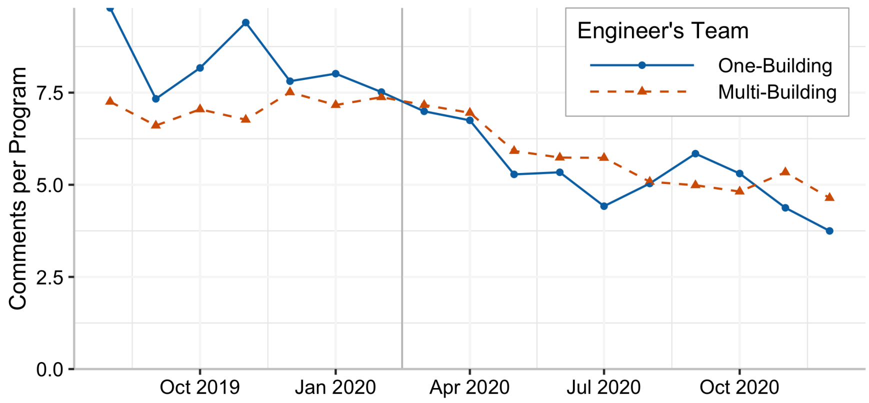
Figure A.7: Robustness of the Effect of Proximity on Online Feedback from Coworkers to Alternative Post-Periods



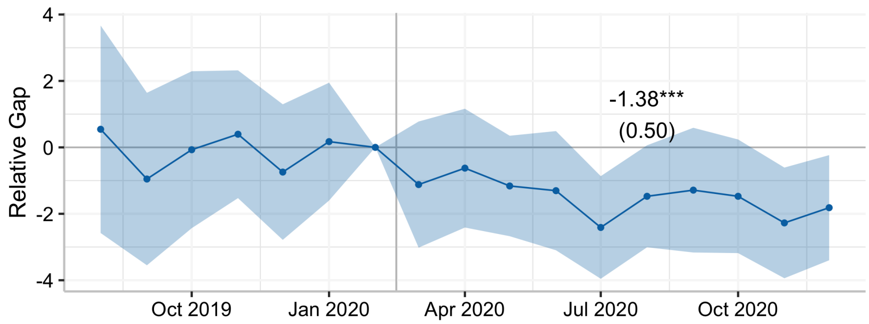
Notes: This figure illustrates how the difference-in-differences estimate from Equation 2 — that compares engineers on one- and multi-building teams, before and after the office closures — varies with the number of months in the post period. The blue circles are the coefficients using our preferred controls for engineering type, engineer tenure, and program scope (in column four of Table 2); the red triangles are the coefficients using the full set of controls (in column six of Table 2). The error bars are 95% confidence intervals with standard errors clustered by engineering team.

Figure A.8: Proximity to Teammates and Online Feedback for Engineers Working on Internal Tools

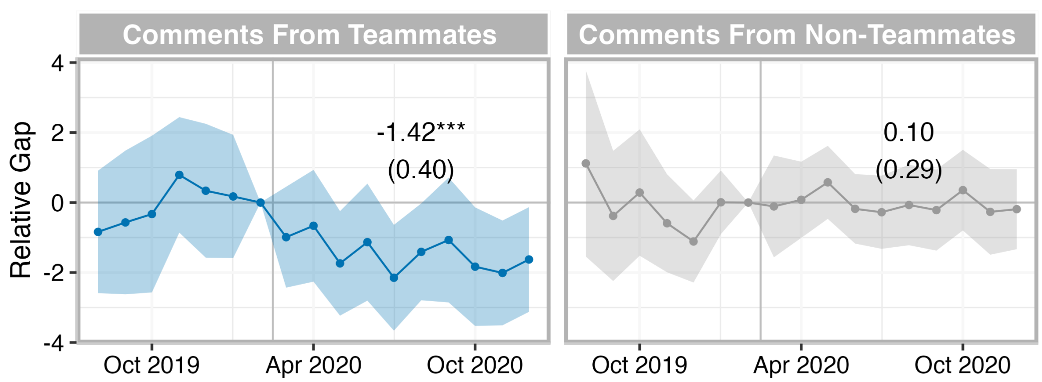
Panel (a): Raw Averages of Comments Per Program



Panel (b): Dynamic, Conditional Differences in Comments Per Program



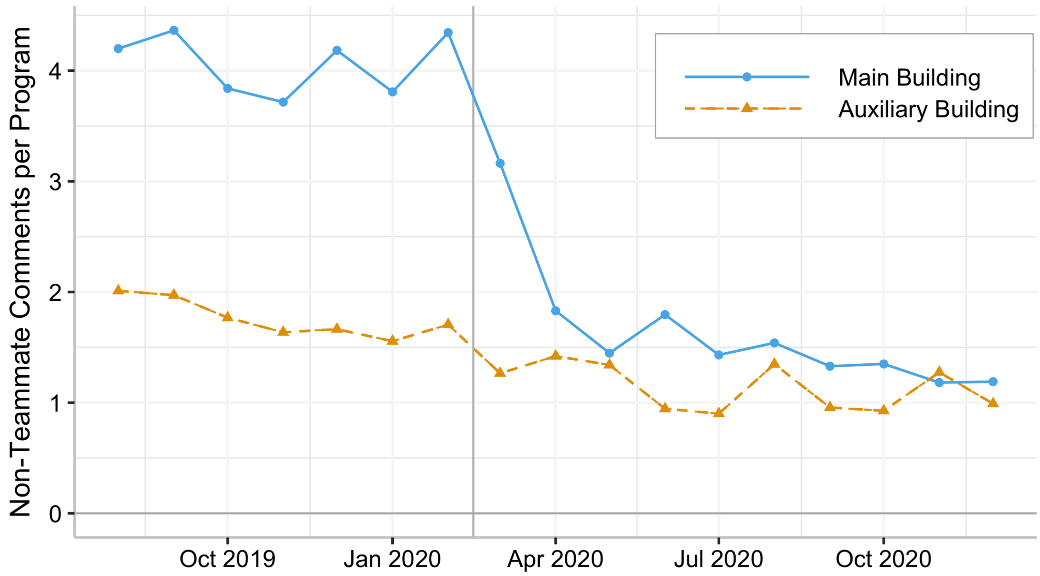
Panel (c): Placebo Check with Comments from Teammates or Non-Teammates



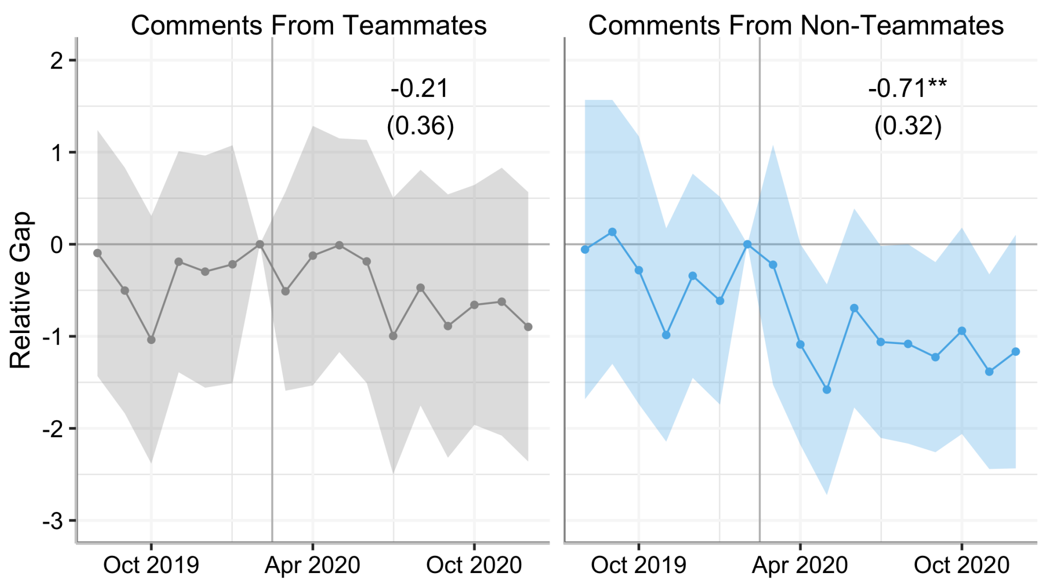
Notes: This figure replicates Figure 1 but limits the sample to the 588 engineers who built internal tools (i.e., software used by others in the firm). Since it is useful for these workers to sit near those who use their tools (e.g., sales workers), it is more likely that their teams end up split across the two office buildings. The analysis compares engineers who had sat with all their teammates in the same building before the offices closed (N=215) to engineers on multi-building teams (N=373). *p<0.1; **p<0.05; ***p<0.01.

Figure A.9: Proximity to Non-Teammates

Panel (a): Raw Averages of Non-Teammate Comments per Program

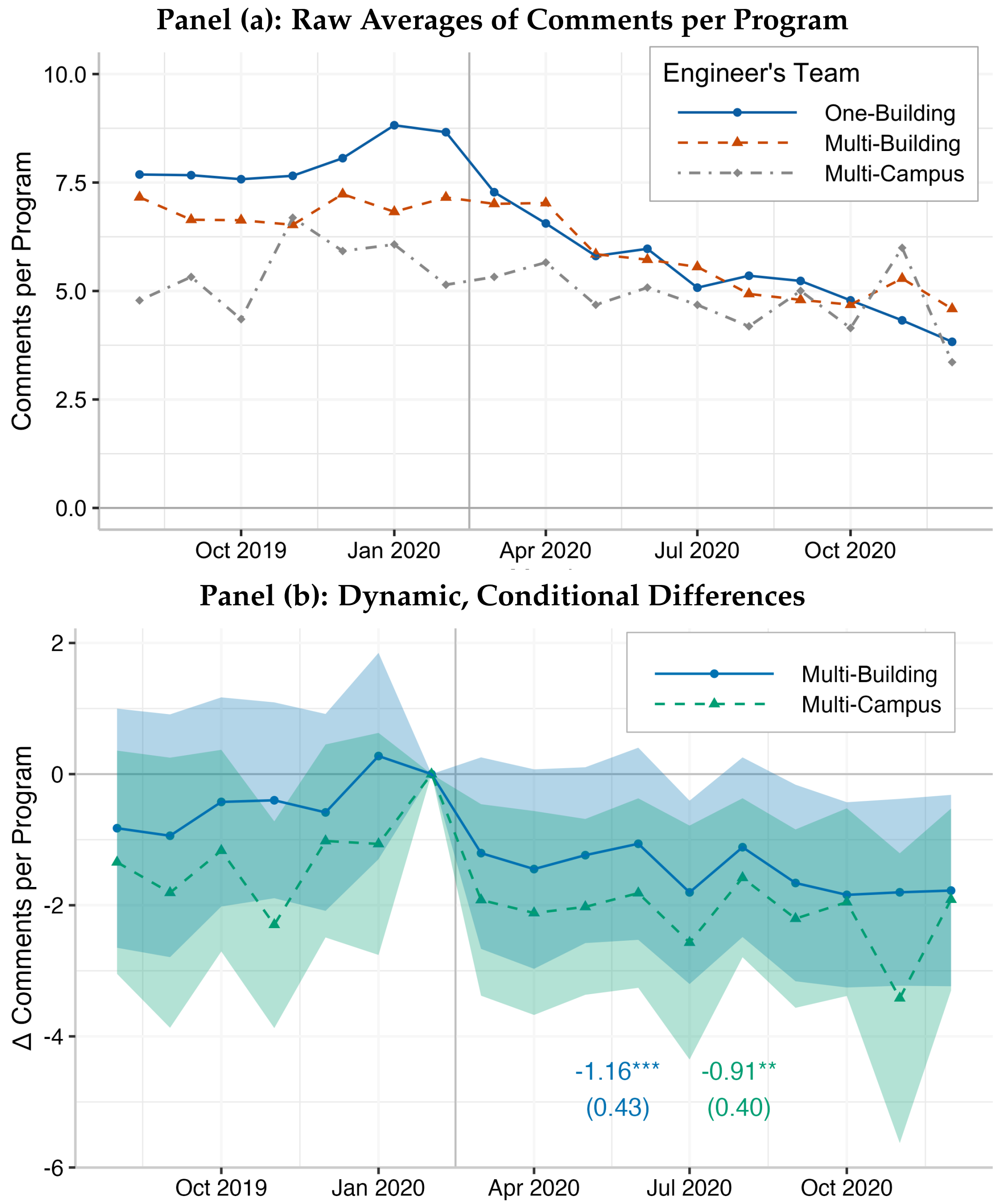


Panel (b): Dynamic, Conditional Differences Placebo and Treated



Notes: This figure compares the feedback received by the engineers who sat in the main building (N=788) and the engineers who sat in the auxiliary building (N=277) around the office closures. Panel (a) plots raw averages in the number of non-teammate comments that engineers receive on their code. Panel (b) presents the conditional differences between engineers in the main and auxiliary buildings, controlling for our preferred controls and the engineers' proximity to their teammates (the analogue of Equation 2 for building rather than team-type). The left plot shows a placebo check with teammate comments. The right plot shows non-teammate comments which should be impacted. Ribbons are 95% confidence intervals with clustering by engineering team. Only engineers whose teammates all worked in the main campus are included. The grey vertical lines mark the COVID-19 office closures. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

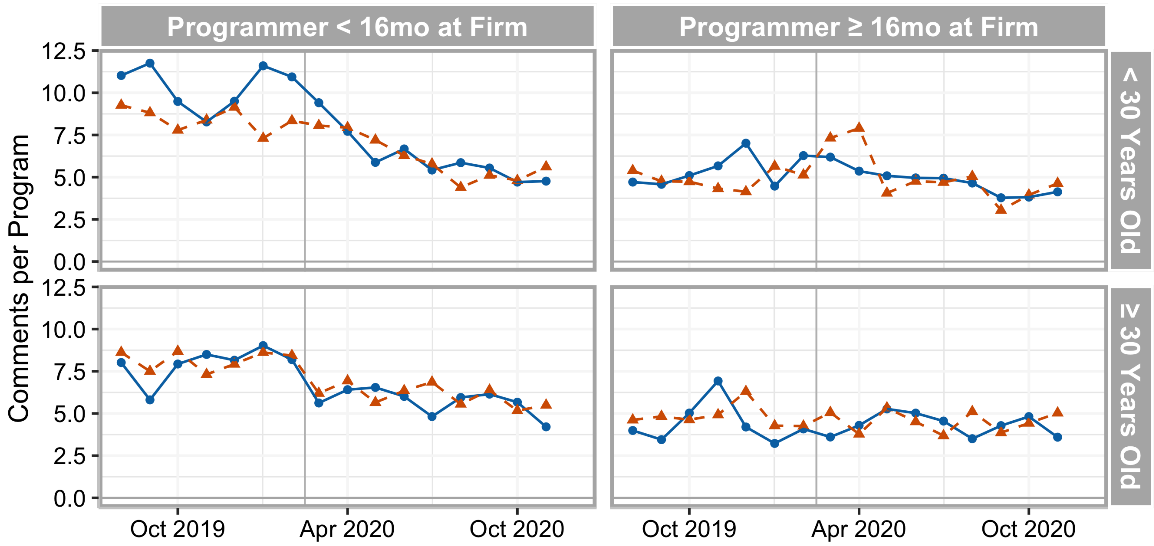
Figure A.10: Proximity to Teammates, by Distance



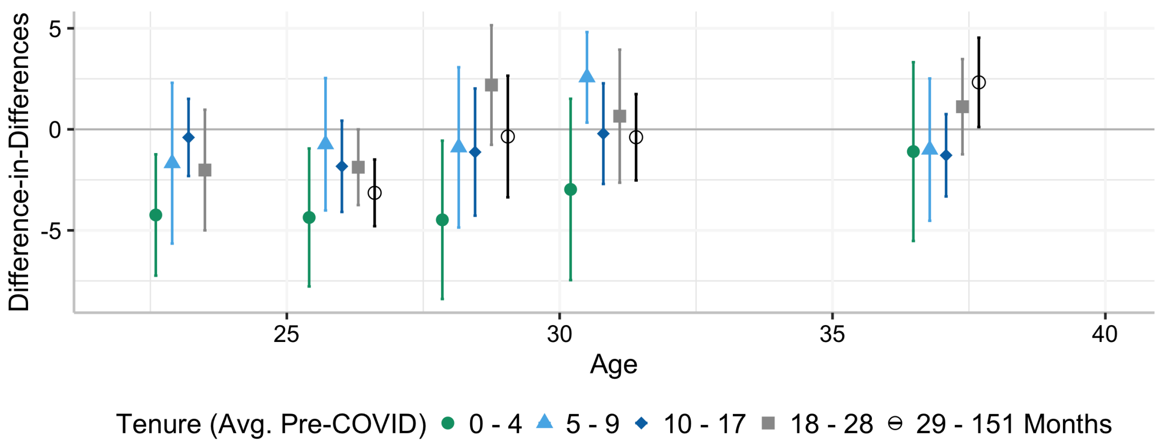
Notes: This figure illustrates the online feedback received by engineers in one-building teams (N=637), multi-building, single-campus teams (N=418), and multi-campus teams (N=215) before and after the offices closed for COVID-19 (the grey vertical lines). The sample includes all engineers who are all themselves in the main campus, regardless of their teammates' locations. Panel (a) plots the raw averages; Panel (b) plots the differences from Equation 3, conditional on our preferred set of controls listed in Subsection III.B. The ribbons reflect 95% confidence intervals with clustering by engineering team. The annotated coefficients come from the analogue of Equation 2. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

Figure A.11: Proximity and Mentorship of Young Engineers

Panel (a): Comments per Program by Program Writer's Tenure and Age



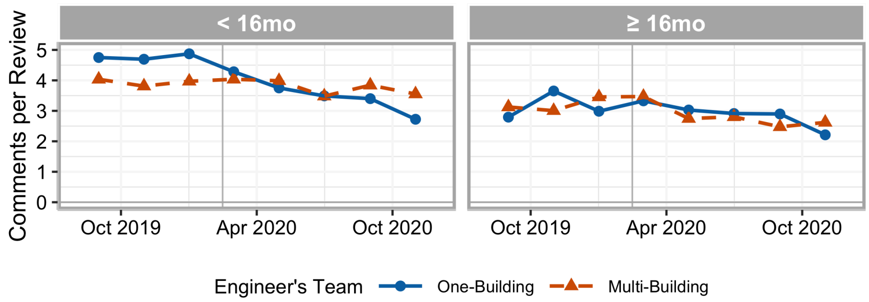
Panel (b): Diff-in-Diff in Comments Per Program by Age and Tenure



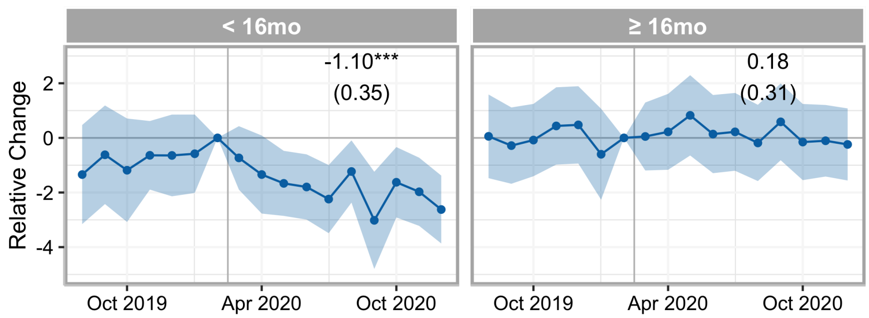
Notes: Panel (a) shows the effects of proximity on online feedback received by engineers of different tenures and ages. It shows the raw monthly averages of comments received per program for engineers on one- and multi-building teams, separately by those below and above the median tenure of 16 months and those below and above the average age of 30. Panel (b) shows the estimated difference-in-differences coefficient from Equation 2 for different age quintiles separately by tenure quintile. Each specification includes our preferred controls for program scope, programmer tenure, and engineering type. Error bars represent 95% confidence intervals with standard errors clustered by team.

Figure A.12: Externalities from Distant Teammates By Tenure

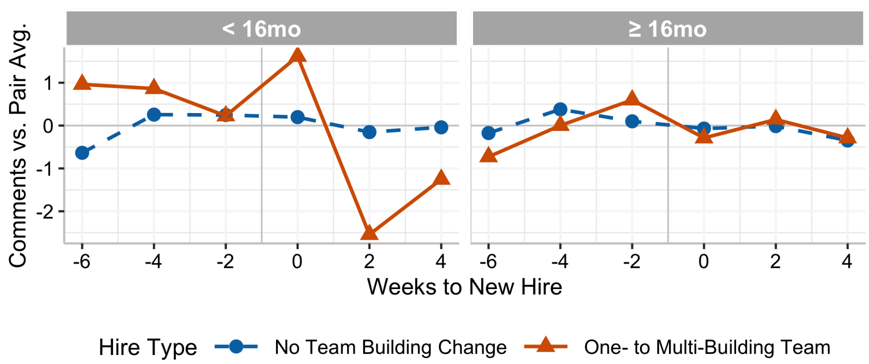
Panel (a): Raw Comments from Same-Building Teammates around Closures



Panel (b): Dynamic, Conditional Differences



Panel (c): Raw Comments/Review from Same-Building Teammate around a Hire

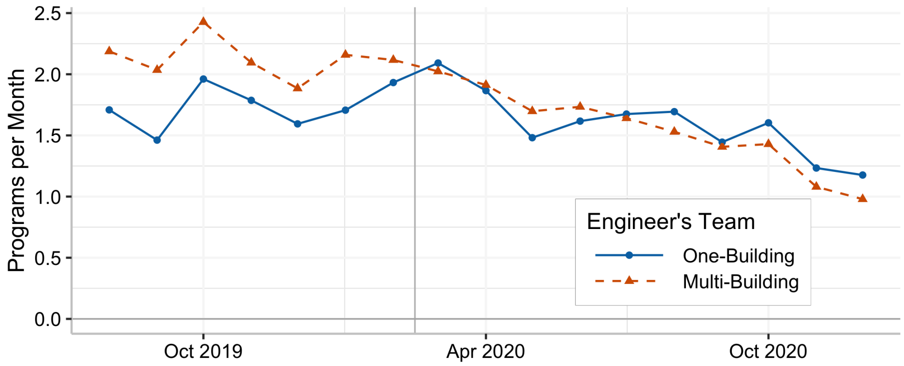


Panel (d): Dynamic, Conditional Differences

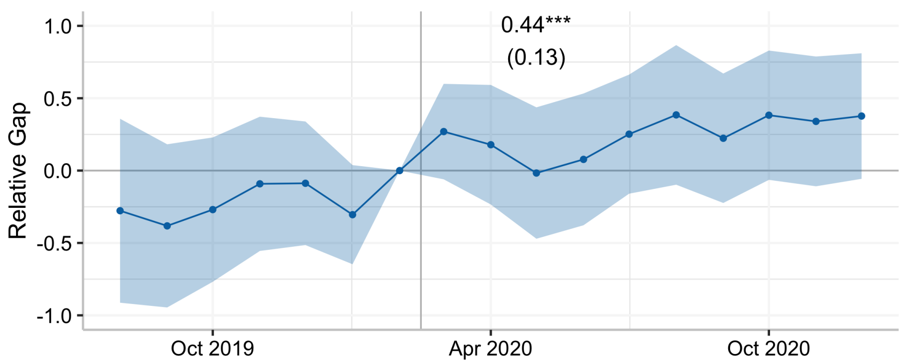
Notes: This figure replicates Figure 3 but differentiates between more and less experienced engineers.

Figure A.13: Proximity to Teammates and Engineer Output for Engineers who Build Internal Tools

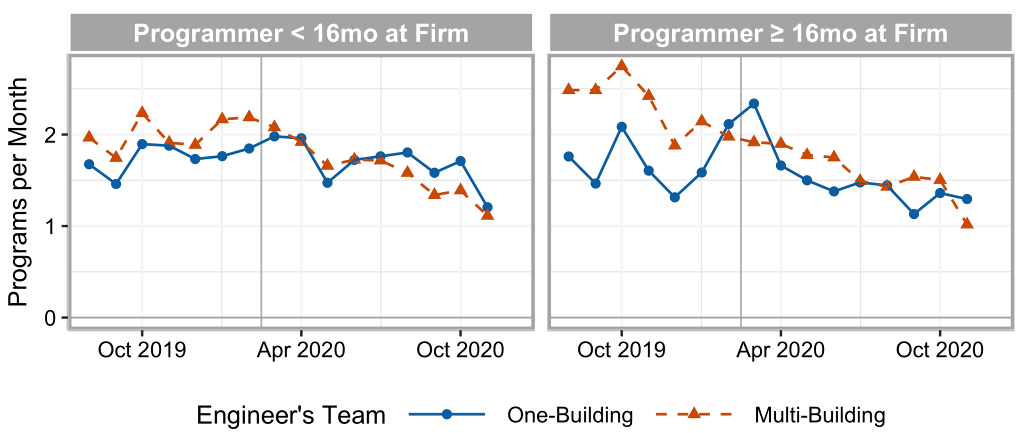
Panel (a): Raw Averages of Programs Written Per Month



Panel (b): Dynamic, Conditional Differences in Programs Per Month

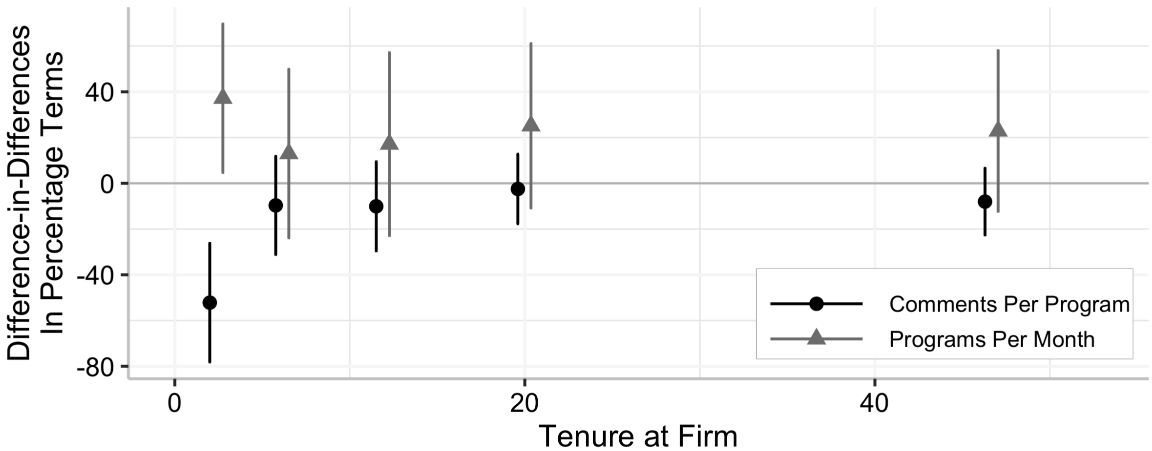


Panel (c): Programs Per Month by Program Writer's Tenure



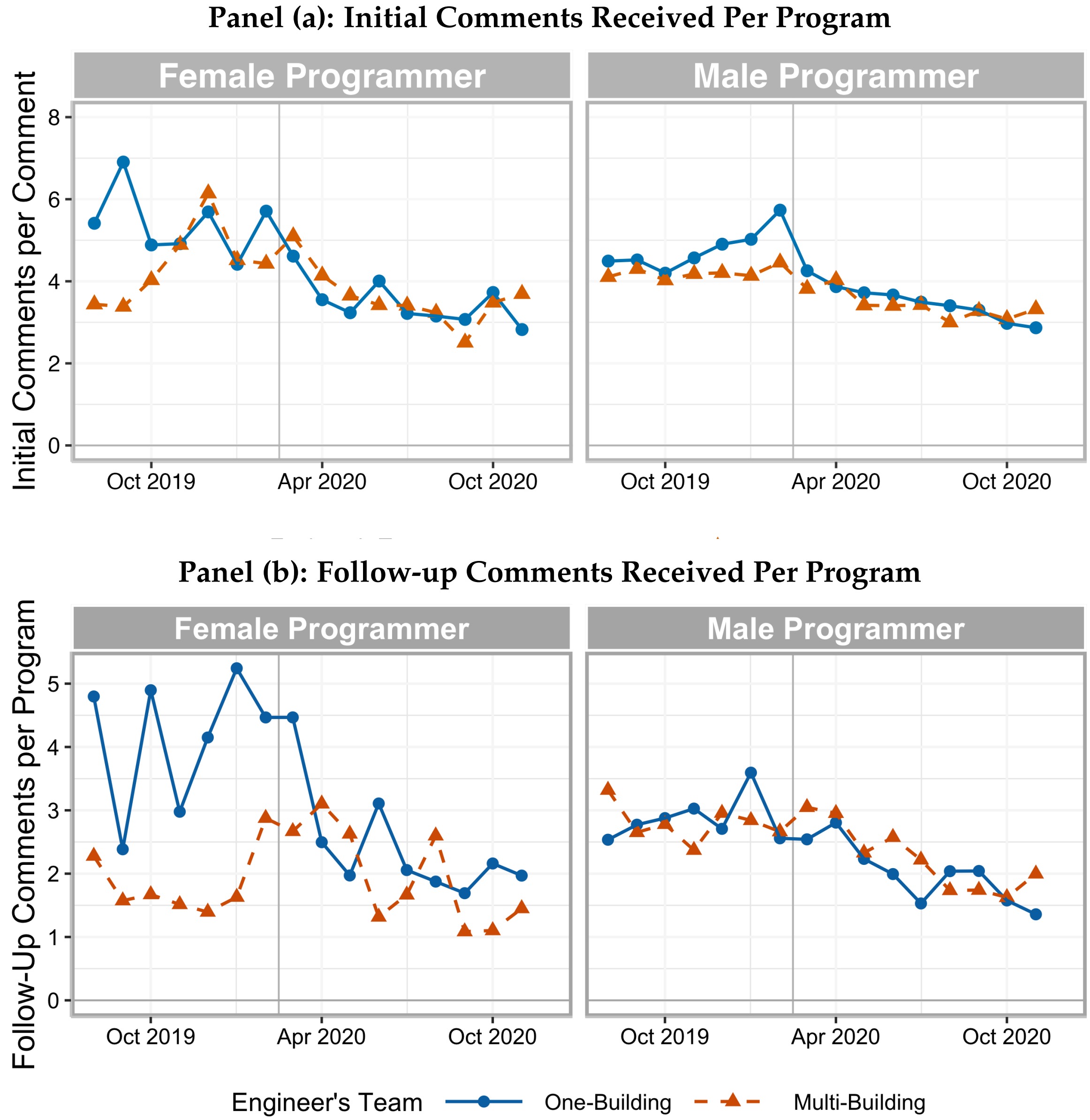
Notes: This figure replicates Figure 4 but limits the sample to the 588 engineers who worked on internal tools (i.e., software used by others in the firm). Since it is useful for these workers to sit near those who use their tools (e.g., sales workers), it is more likely that these teams are split across the two office buildings. The analysis compares engineers who had sat with all their teammates in the same building before the offices closed (N=215) to engineers on multi-building teams (N=373). Standard errors are clustered by engineering team. *p<0.1; **p<0.05; ***p<0.01.

Figure A.14: Proximity to Teammates and Engineer Output and Feedback by Baseline Tenure



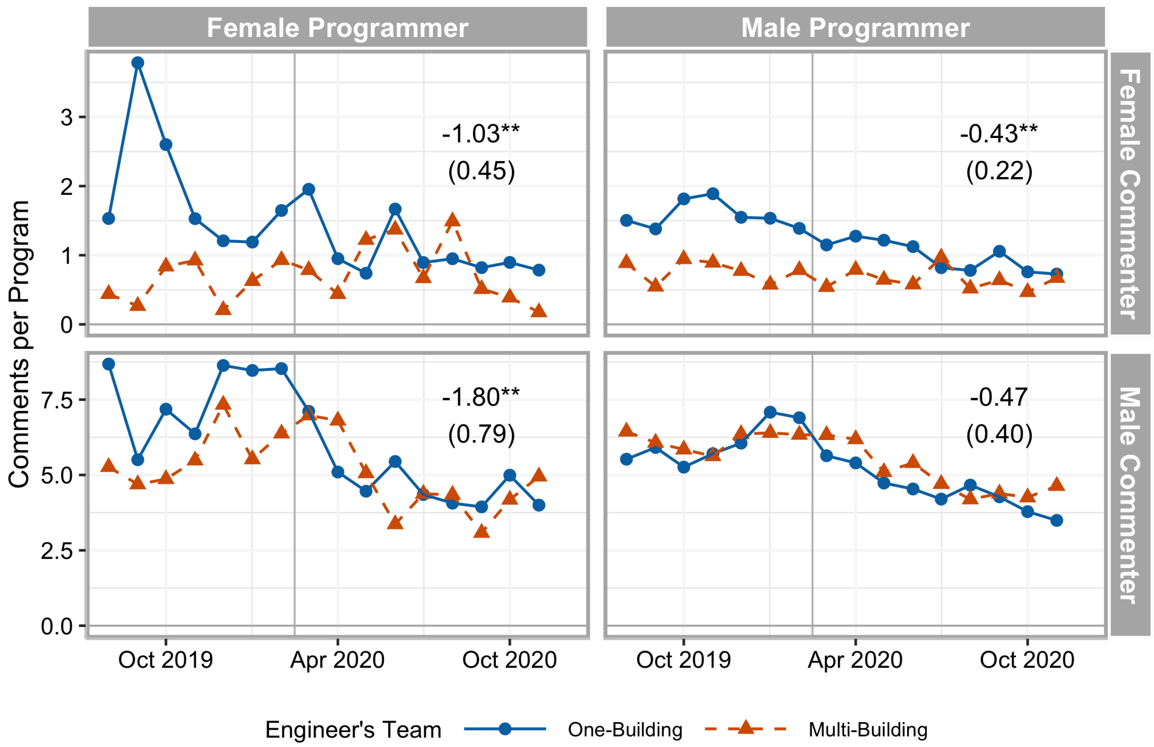
Notes: This figure illustrates the effects of losing proximity to teammates on feedback and programming output for engineers of different tenure at the firm. Each point represents a quintile of baseline tenure at the time that the offices closed. The y-axis plots the difference-in-differences coefficient from Equation 2 with our preferred set of controls for the scope of the program (quartics in files changed, lines added, and lines deleted), the engineer's tenure at the firm (in months) and the engineering role (e.g., website design versus database management). The error bars represent 95% confidence intervals with standard errors clustered by team.

Figure A.15: Gendered Impacts of Proximity to Teammates on Initial and Follow-up Feedback



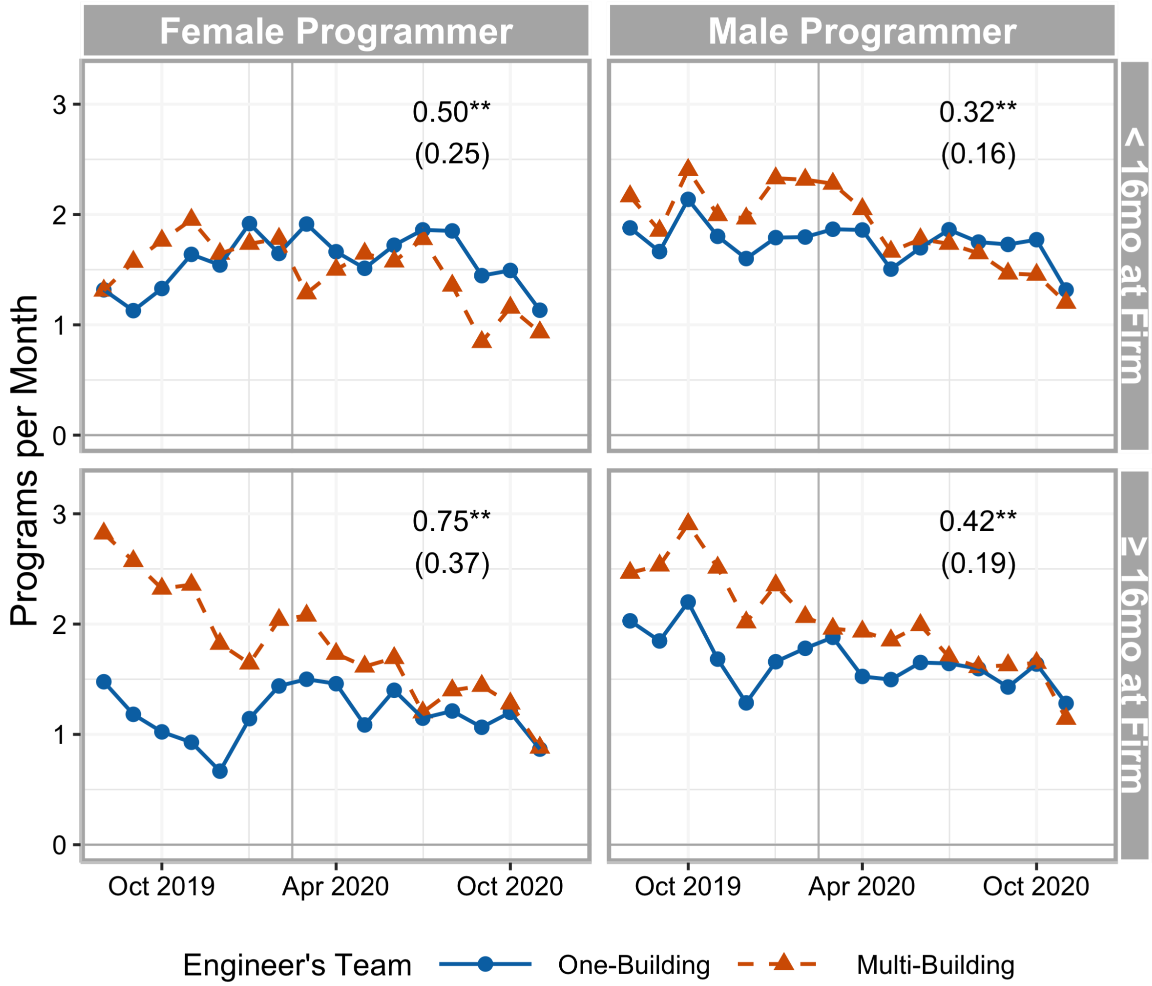
Notes: This figure illustrates the gendered impact of proximity on the feedback that an engineer receives on their code. Panel (a) plots the initial feedback on their code, defined as the average number of comments that an engineer receives on their programs before they send a follow-up reply. Panel (b) plots the average number of comments that an engineer receives on their code after they send a reply. The sample limits to engineers whose teammates all worked in the main campus.

Figure A.16: Gendered Effects of Proximity on Comments from Male and Female Commenters



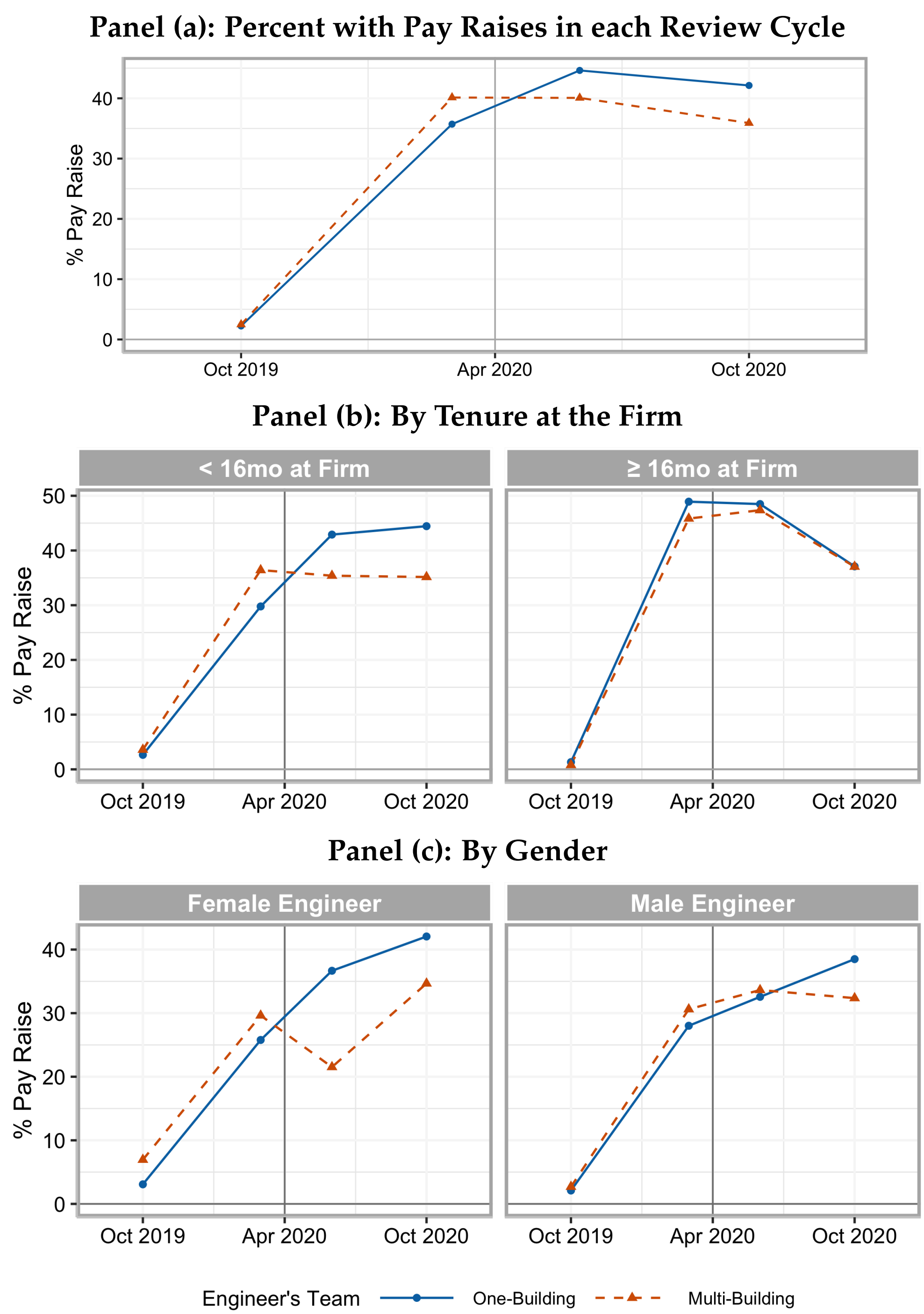
Notes: This figure illustrates the gendered impact of proximity on feedback from male and female commenters. Each plot shows the raw monthly averages of comments received per program for engineers on one- and multi-building teams, separately by female and male engineers and for comments from male and female commenters. The sample limits to engineers who submitted a program to the main code-base in that month and whose teammates all worked in the main campus. The annotated coefficient reflects Equation 2 with our preferred set of controls for program scope, engineering type, and engineer tenure. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Figure A.17: Gendered Impacts of Proximity to Teammates on Programming Output by Seniority



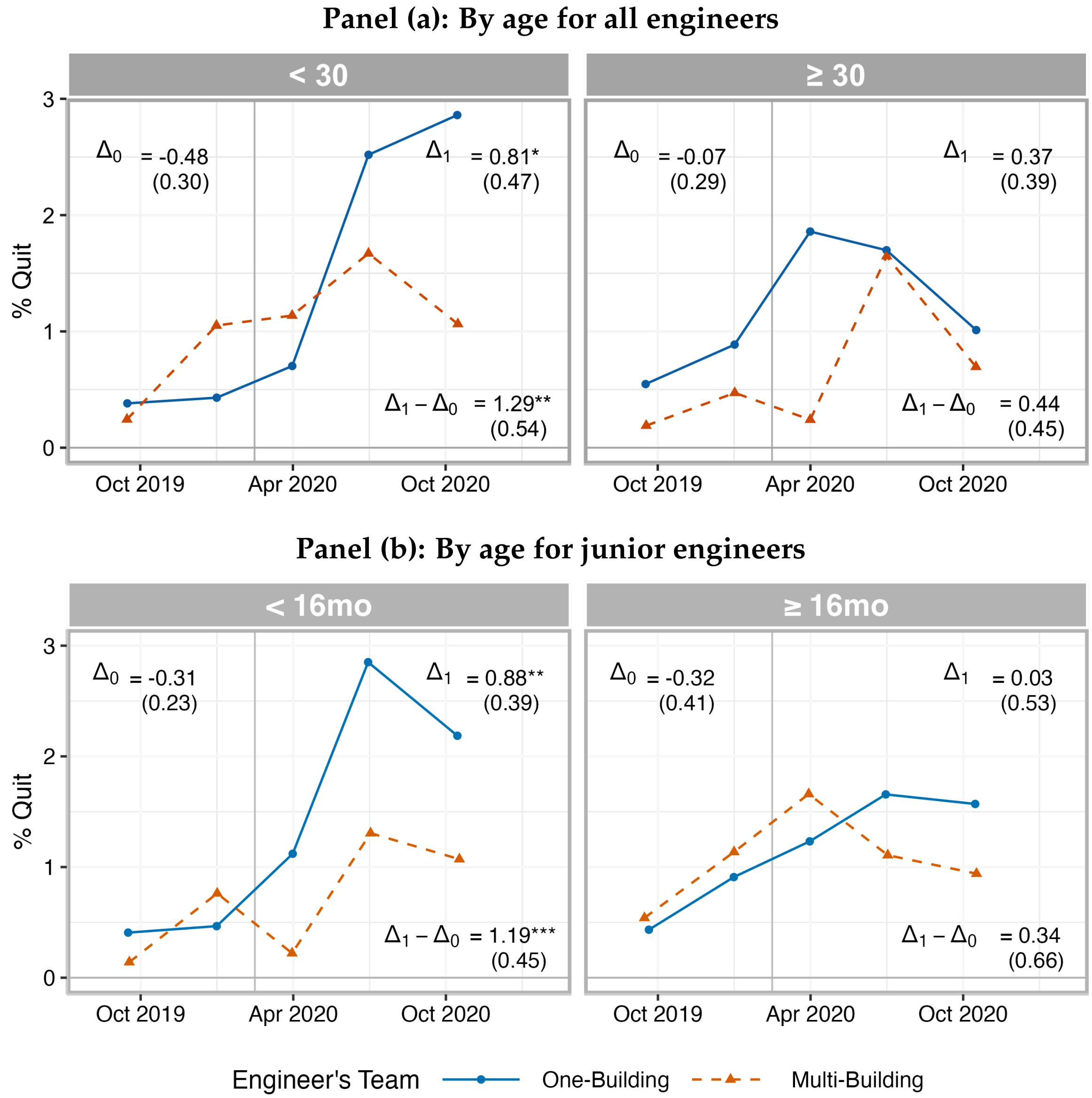
Notes: This figure illustrates the gendered impact of proximity on programming output. Each plot shows the raw monthly averages of programs per program for engineers on one- and multi-building teams, separately by female and male engineers and for those with above and below the median tenure of 16 months. The sample limits to engineers whose teammates all worked in the main campus. The annotated coefficient reflects Equation 2 including our preferred set of controls for engineering type and engineer tenure and project scope. Standard errors are clustered by team. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

Figure A.18: Impacts of Proximity on Pay Raises



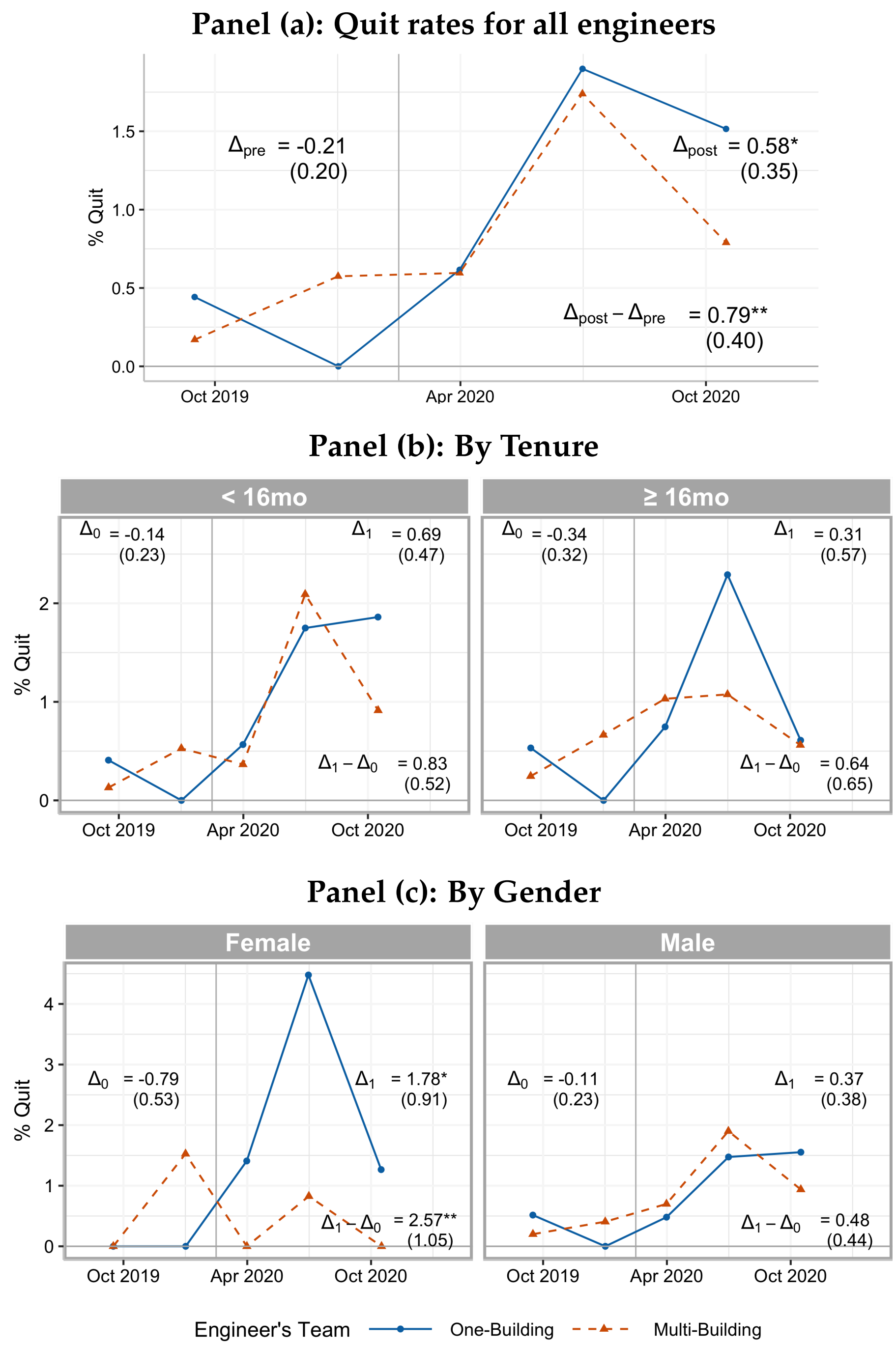
Notes: This figure illustrates the impact of proximity on the likelihood of a pay raise. Each point reflects the percent of engineers with pay raises at the end of each tri-annual review period. The post period is defined as starting in April 2020 since March 2020 pay raises were based on winter 2019-2020 reviews. The sample includes engineers who worked on the main code-base, whose teammates were all in the main campus, and who were hired before the start of our sample.

Figure A.19: Impacts of Proximity on Quits by Age



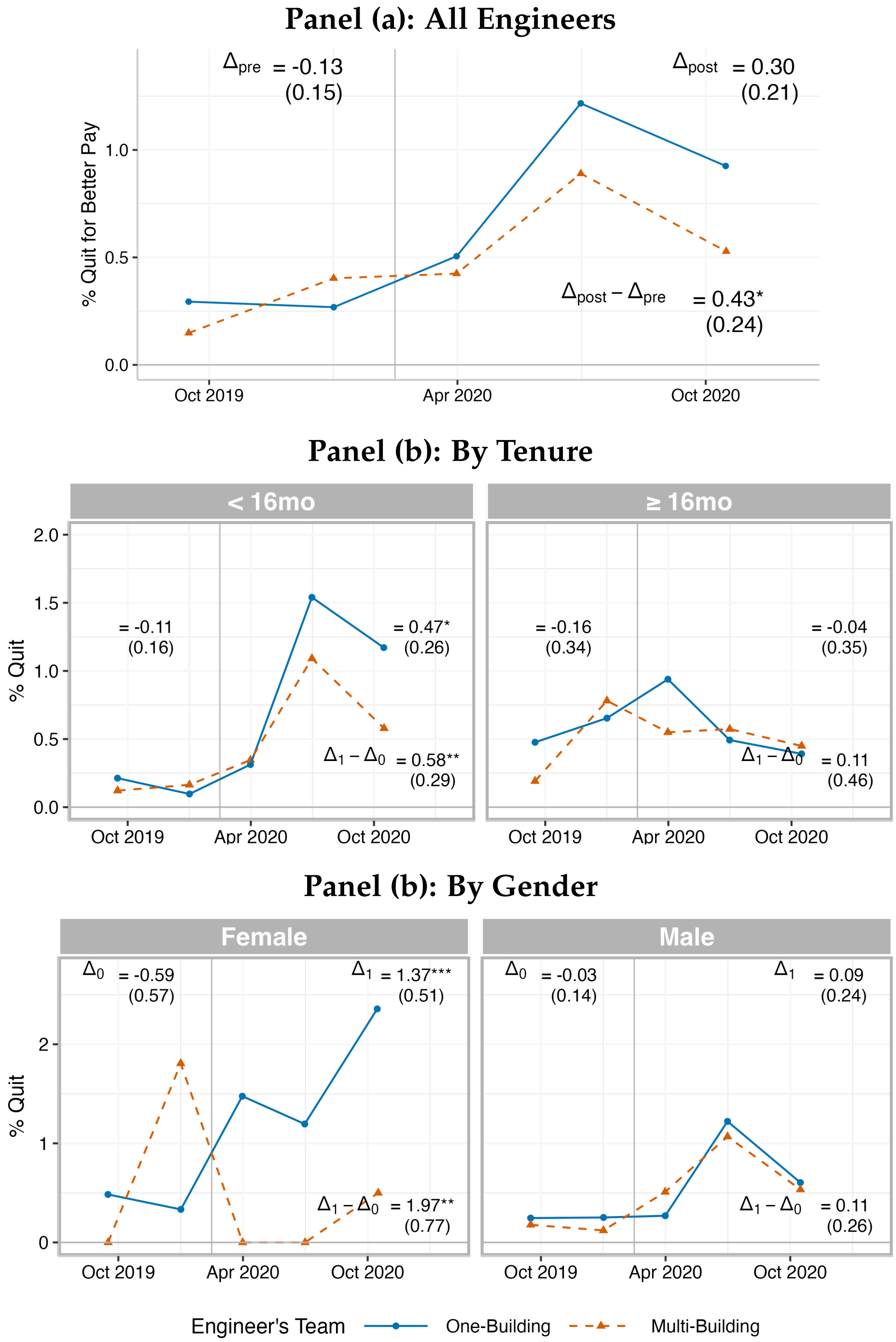
Notes: This figure illustrates the effects of proximity on quits for older and younger engineers (a) overall and (b) for relatively junior engineers before COVID (with less than the median sixteen months of experience). Each plot shows the raw quit rates for engineers on one-building and multi-building teams. The annotated coefficients use our preferred set of controls for engineering group and engineer tenure. Standard errors are clustered by engineering team. The sample includes engineers who worked on the main code-base, whose teammates were all in the main campus, and who were hired before the start of our sample. *p<0.1; **p<0.05; ***p<0.01.

Figure A.20: Impacts of Proximity on Quits for Engineers who Work on Internal Tools



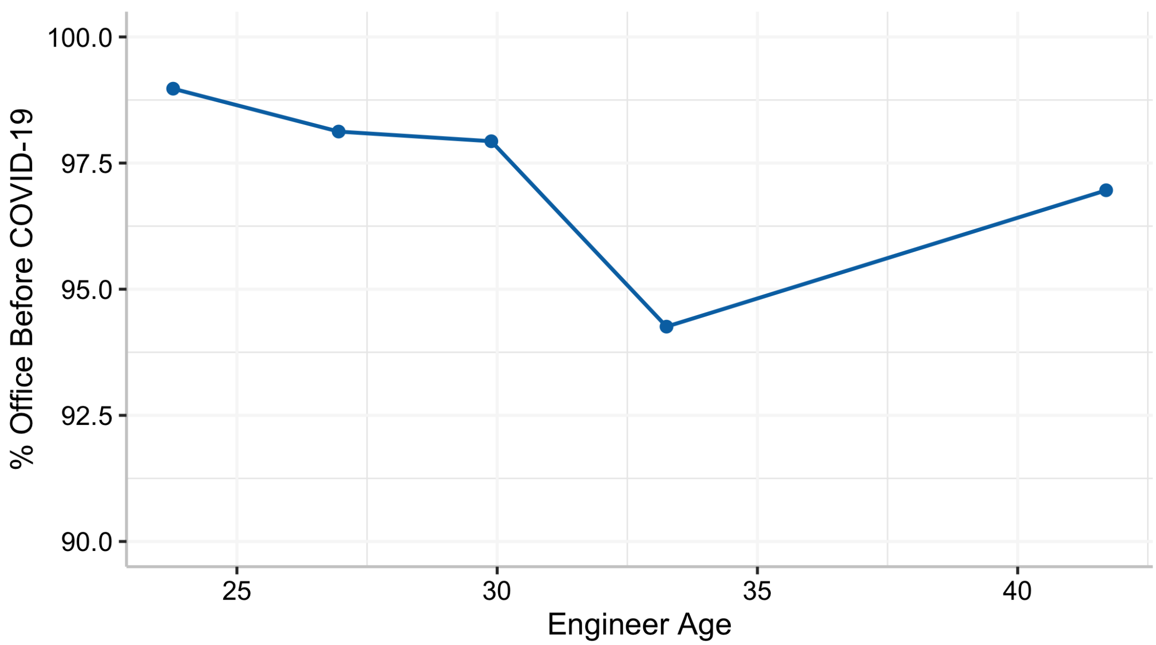
Notes: This figure replicates Figure 6 but limits the sample to the 588 engineers who worked on internal tools. Since it is useful for these workers to sit near those who use their tools (e.g., sales workers), it is more likely that these teams are split across the two office buildings. The analysis compares engineers who had sat with all their teammates in the same building before the offices closed (N=215) to engineers on multi-building teams (N=373). Standard errors are clustered by engineering team. *p<0.1; **p<0.05; ***p<0.01.

Figure A.21: Impacts of Proximity on Quits for Higher Paying Jobs



Notes: This figure replicates Figure 6 focusing on quits to jobs with higher average compensation on Glassdoor than the current firm's average salaries on this site. We compare workers who quit for higher pay to workers who did not quit or quit for a position with lower average compensation. We exclude workers who quit but for whom we do not know the destination compensation. The sample includes engineers who worked on the main code-base, whose teammates were all in the main campus, and who were hired before the start of our sample. The annotated coefficients use our preferred set of controls for engineering group and engineer tenure. Standard errors are clustered by engineering team. *p<0.1; **p<0.05; ***p<0.01.

Figure A.22: Office Work at the Firm by Engineer Age



Notes: This figure plots the share of engineers at the firm who were working in the office rather than from home before the pandemic as a function of the engineer's age.

Table A.1: Example Emblematic Comments of Principal Components

PCA	Total Load	Text of Comment
1	1.25	This **logic** is very confusing to **read** for the first **time**. Please add a paragraph in the comment section to clearly explain the **logic**.
1	0.98	So, you are still **reading** all jobs and then **counting** in PHP. I would think you want to make a **count** query to **SQL server**.
1	0.90	If you change it to **SELECT COUNT**(*) AS **COUNT** without a GROUP BY, would that work for you? GROUP BY is an expensive operation on our **server**
1	0.87	Do we want to add the price range **logic** in there then? I'm confused why that **logic** hasn't changed but this **logic** here has.
1	0.85	Do you think it's getting close to time to separate the filtering **logic** and the transfor ming **logic** here?
2	1.29	We agree and approve the **ownership** transfer from Jane Doe's **team** to our **team**.
2	1.08	Can you please get **approval** from John Doe's **team** first? We have a pr in review which makes John Doe's **team codeowners** of this file.
2	0.95	Instead of giving static **team** id, can we give random **team** id. Get **team** id from tabl e and do ORDER BY `ORDER BY newid()` to get random **teams**
2	0.92	If this **team** is yours, it would be great if you could migrate the - **owners** in this change to the appropriate **owner team** and **reviewer** group
2	0.85	LGTM note: should we move this file **ownership** in **codeowners** to your **team**?
2	−0.48	This if **return**, else if **return** else **return** statement can be turned into if **return**, if **re turn**, **return**
2	−0.41	Type hint **return** int, if it is **nullable** you can use ?int. ``` **public function** id_supplier(): int { **return** $this->id_supplier; } ```
2	−0.39	Can we just **return** this **bool** check? Rather than if () {**return false**} else {**return true**}
2	−0.38	Stick this in the **return** below so we have `**return** a || b || c;` instead of `if(a) {**retur n** true}; **return** b || c;`
2	−0.38	Should update this **return** type to match the actual **return** type. Especially since you added a **return** type hint.
2	−0.38	Could we add the **return** type hint? Also in the doc string above, the **return** is listed as **string** but this **function** returns a **function**.
3	1.06	Why do we want to do this? If a **class** has a **test class** and non-**test class**, it should be able to run the **test** file, no?
3	0.99	Could we create a **function** in conversation model and put the **DAO** function there? Then here, we'll call the **model function**.
3	0.91	Any reason to make this as a static **function** and moving it to **DAO**? Looks more like an helper **function** instead of a **DAO function**?
3	0.88	If this **method** and the **method** below are only used in **test**, could we add them in th e **test DAO class**?
3	0.86	Make this into a wrapper around the real **method**, and pass the request into the real **method**. Then you can create a unit **test** for that **method**.

Notes: This table shows five emblematic examples with high loadings on each principal component. The second principal component has identified two clusters of comments: where each cluster includes words that often appear together but rarely appear with words in the other cluster. These comments were selected because they have the maximum loadings of all comments less than 150 characters long. See Section I.A for details on principal component analysis.

Table A.2: Example Comments Evaluated by Software Engineers

Comment	Helpful?	Change code?	Actionable steps?	Explain reasoning?	Rude?
nitpick: If would have been nice if you designed this so that we can have the table in a place other than csn_service. But that's ok if you don't want to do this for this ticket.	Yes	No	Yes	Yes	No
clean up this comment	Not enough info	Not enough info	No	No	A little bit
Are you certain once the variables are binded to the SQL string that the null variables aren't going to get casted as string and result in NULL which wouldn't be null? i.e. \`THEN ISNULL(NULL, 0)\`	Yes	No	No	Yes	No
A general comment that should be applied everywhere. We should be keeping the naming of these entities consistent. Instead of orderSplits and orderItemSplits we should be calling things inventoryOrderSplits and inventoryOrderItemSplits.	Yes	Yes	Yes	Yes	No
This maybe required	Yes	Yes	Yes	Yes	A little bit
IIRC you can update the collection class phpdoc so that PHPStorm can detect the model type from \`getDeliveryZones()\` automatically instead of writing it manually ^ - [link]	Yes	Yes	Yes	Yes	No
Looks like the key 'must_not_ship_before_date' is used in a couple of places, here and shipping_label_helper.php is it possible to create a constant for it somewhere and reuse it?	Yes	Yes	Yes	No	No
What does this look like on the frontend, will it just be a blank view? Can we make it just show an error message ,something along the lines of Failed to generate preview. Please click 'Preview Creative' to try again? If that is possible/easy?	Yes	Not enough info	Yes	Yes	No
Function description	Yes	Yes	Yes	Yes	No
This should be changed back to \`getNavHeaderHtml()\`	Yes	Yes	Yes	Yes	No
Leave all these properties unset and instead make them required in the constructor as parameters. Then child constructs can just	Yes	Yes	Yes	Yes	No

Notes: This table continued on the subsequent pages shows examples that were rated by software engineers.

pass them in the `parent::` call (while the dependencies like `App Nexus Client` and `DAO` can be injected)					
Good improvement I think.	Yes	No	No	No	No
Capitalize EXISTS	Yes	Yes	Yes	No	A little bit
can you import the logger to not use the fully qualified name?	Yes	Not enough info	Yes	No	A little bit
`'web'` probably should be a constant	Yes	Yes	Yes	Yes	No
This doesn't seem right -- why are we defaulting to `Feature::is_enabled` instead of an instance of `Toggle`? Shouldn't we be avoiding static calls to `Feature` in this manner?	Yes	Yes	Yes	Yes	No
Are SOP's going to have any idea how to handle an error thrown from a try/catch block/will they have any action they can take after one of these errors? Might be better to just log this and display a generic error rather than the specific one.	Yes	Yes	Yes	Yes	No
This guy wants to be an object	Not enough info	Not enough info	Not enough info	Not enough info	A little bit
Hm, there's no `ORDER BY` on the DAO call here, so maybe the limit solution doesn't actually work, since you may just return the same `N` tran_logs every time...We'd have to add date-based looping combined with was there at least N returned from the last set to ensure we've validated every tran log we need to look at has been. What's the current rate for 158's per day? It may be worth doing a loop over each day or couple days	Not enough info	Not enough info	Yes	Yes	No
nit: extra white space	Yes	Yes	Yes	No	No
Why ISNULL() here? I believe in your :delivery_date and :ship_date will never actually be null, and imo it makes the fn less intuitive if someone tries to set the value of one of these fields to NULL	Yes	Yes	Yes	Yes	No
might also want to check for an actual null value here, not just the word 'null' or an empty string	Yes	Yes	Yes	Yes	No
Slight nitpick, but is `skuName` set in stone as a requirement? It seems like	Yes	Yes	Yes	Yes	No

`productName` should map to `$product-name` and `displayName` should map to `$product-display_name`, for example (and consistency with the product models)					

What if we keep the cartons with no SSCC and remove the cartons with no SSCC in the `serialize` function? Or implement a validate function on the model and act on it according to what the validate function says - [link]
 	
``
if (!$collection-validate_models()) return array_filter($collection, 'exclude_empty_sscc'))
return $collection
``
 	
This way, we still keep cartons with no sscc in the collection. And don't have to guess any point in our code if we've removed cartons with no sscc or not.
	Yes	Yes	Yes	Yes	No

nice!!	No	No	No	No	No
I know this is how the current faux joins do it... but why do we fill an array with null?	Yes	No	No	No	No
I assume the commented out client logger and stub is left over from local troubleshooting?	Yes	Not enough info	No	Not enough info	Moderat ely
should we add info level kibana log here	Yes	Yes	No	No	No
Should we keep the `- Zones_To_Pick[]` since we already have it as the param type?	Yes	Yes	Yes	Yes	No
LGTM: Only formatting comments for me	Yes	Yes	Yes	Yes	No
I'm wondering if this is more complicated than necessary and if we instead should be taking advantage of the functions already tried and true in the bulk helper file	Yes	Yes	Yes	Yes	No
HTTP errors are logged in L1926. This code is only for timeouts	Yes	Yes	Yes	Yes	No
Change year	Yes	Yes	Yes	Yes	No
WF3 sniff seems to be failing. Can you fix this?	Yes	Not enough info	No	No	No
Add more info on description/testing done. Run ```firm sniff``` / ```firm format``` and address issues: ```	Yes	Yes	Yes	Yes	No

FILE: .../wms/dimension_update_tool/dimension_update_tool_detail_view.php -- *emojii* | -- *emojii* | FOUND 3 ERRORS AFFECTING 2 LINES *emojii* | -- *emojii* | 19 [link]| fully-specified classpath *emojii* | -- ```					
another nit: don't love the name here, because it will _have_ to be changed if the test wins	No	Not enough info	No	No	No
just wanted to make sure Adam is warned of this change	No	Not enough info	No	No	No
our sniffer prefers `random_int()` instead of `rand`	Yes	Yes	Yes	Yes	No
nit: can remove `public` here	Not enough info	Not enough info	Yes	No	No
Agree with John here, those store constants don't exist in every context and using them has got us into trouble in the past (looking at you, Category Builder)	Yes	Yes	Yes	Yes	No

Table A.3: Gender Gaps in Online Feedback: Coded Comments

Panel (a): Overall gender gap

	% Of Comments									
	Helpful		Change Code		Actionable		Explain Reason		Rude	
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
Female	2.90**	3.56***	2.41*	2.97**	2.35*	2.58*	−0.22	0.63	−0.71	−0.72
	(1.14)	(0.92)	(1.41)	(1.45)	(1.37)	(1.36)	(1.36)	(1.41)	(1.08)	(1.12)

Panel (b): Gender gap by team-type and period

	Helpful		Change Code		Actionable		Explain Reason		Rude	
Female										
One-Building, Pre	2.03	4.20**	3.50	5.82**	1.79	3.40	0.78	3.75	0.53	0.76
	(2.37)	(2.10)	(2.80)	(2.80)	(2.69)	(2.48)	(2.50)	(2.65)	(2.32)	(2.36)
Multi-Building, Pre	1.85	4.18*	0.27	−0.77	−0.56	0.91	−2.94	−0.95	−1.69	−1.48
	(2.54)	(2.22)	(2.86)	(3.25)	(2.83)	(2.77)	(3.11)	(3.21)	(2.05)	(2.13)
One-Building, Post	4.23*	3.81**	1.50	1.32	2.87	3.35	−0.85	−0.23	−1.69	−1.96
	(2.29)	(1.74)	(2.88)	(3.07)	(2.84)	(3.04)	(2.55)	(2.81)	(2.10)	(2.17)
Multi-Building, Post	3.97	2.28	5.26*	6.68**	6.06**	3.12	2.87	0.36	0.13	−0.18
	(2.48)	(1.85)	(2.85)	(2.96)	(2.91)	(2.86)	(2.63)	(2.64)	(1.80)	(1.87)
Preferred Controls	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Drop Not Enough Info		✓		✓		✓		✓		✓
Dependent Mean	76.2%	87.0%	52.0%	70.4%	60.3%	68.4%	51.4%	58.4%	14.7%	15.2%
# Comments	5,377	4,708	5,377	3,974	5,377	4,741	5,377	4,740	5,377	5,174
# Teams	258	255	258	253	258	254	258	254	258	258

Notes: This table evaluates the gender gap in feedback for the random subset of comments that engineers evaluated on their helpfulness and content. Panel (a) considers the overall gender differences with our preferred set controls. Panel (b) allows the gender difference to vary by the team-type (one- versus multi-building) both before and after the office closed (pre vs. post). Standard errors are clustered by team. See the note of Table A.7 for the definitions of the outcomes. *p<0.1; **p<0.05; ***p<0.01.

Table A.4: Experience Gaps in Online Feedback: Coded Comments

Panel (a): Overall experience gap

	% Of Comments									
	Helpful		Change Code		Actionable		Explain Reason		Rude	
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
Junior (< 16mo)	3.40***	1.59	4.90***	6.41***	6.66***	6.62***	4.27***	4.40***	−2.40**	−2.43**
	(1.23)	(1.07)	(1.46)	(1.61)	(1.61)	(1.64)	(1.56)	(1.63)	(1.04)	(1.08)

Panel (b): Experience gap by team-type and period

	Helpful		Change Code		Actionable		Explain Reason		Rude	
Junior (< 16mo)										
One-Building, Pre	5.07*	1.47	4.68	5.43	3.08	2.07	0.29	−0.15	−3.36	−3.39
	(2.73)	(2.31)	(3.11)	(3.33)	(2.74)	(2.81)	(3.01)	(3.09)	(2.63)	(2.67)
Multi-Building, Pre	4.30	2.39	5.35	4.46	9.32**	9.86***	6.63*	6.21*	−1.76	−1.41
	(2.90)	(2.33)	(3.62)	(3.73)	(3.68)	(3.48)	(3.42)	(3.42)	(2.07)	(2.17)
One-Building, Post	1.54	1.88	3.48	4.92*	5.42**	5.36**	1.94	2.84	−2.13	−2.15
	(2.29)	(1.72)	(2.60)	(2.74)	(2.54)	(2.68)	(2.70)	(2.79)	(1.82)	(1.94)
Multi-Building, Post	2.50	0.13	5.15**	9.01***	7.22***	7.66***	6.33**	6.82**	−2.46	−2.76
	(2.10)	(1.96)	(2.14)	(2.84)	(2.65)	(2.74)	(2.86)	(3.16)	(1.84)	(1.90)
Preferred Controls	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Drop Not Enough Info		✓		✓		✓		✓		✓
Dependent Mean	76.2%	87.0%	52.0%	70.4%	60.3%	68.4%	51.4%	58.4%	14.7%	15.2%
# Comments	5,377	4,708	5,377	3,974	5,377	4,741	5,377	4,740	5,377	5,174
# Teams	258	255	258	253	258	254	258	254	258	258

Notes: This table evaluates the experience gap in feedback for the random subset of comments that engineers evaluated on their helpfulness and content. Panel (a) considers the overall differences between junior engineers with less than the median tenure at the firm (16 months) and senior engineers with more experience. We include our preferred set of controls, sans the usual tenure controls, which would render the junior indicator not identified. Panel (b) allows the experience difference to vary by the team-type (one- versus multi-building) both before and after the office closed (pre vs. post). Standard errors are clustered by team. See the note of Table A.7 for the definitions of the outcomes. *p<0.1; **p<0.05; ***p<0.01.

Table A.5: Testing Robustness of Results to Local-Linear Time-Trends

	Comments per Program					
	(1)	(2)	(3)	(4)	(5)	(6)
Post x In One-Building Team	-1.29^{***}	-1.72^{**}	-1.16^{***}	-1.41^{**}	-1.14^{**}	-1.36^{**}
	(0.48)	(0.86)	(0.43)	(0.63)	(0.47)	(0.64)
One-Building Team	1.16^{**}	1.82^{**}	1.80^{***}	2.50^{***}		
	(0.52)	(0.93)	(0.49)	(0.66)		
Post	-1.22^{***}	-0.24				
	(0.36)	(0.63)				
Pre-Mean in One-Building Teams	8.04	8.04	8.04	8.04	8.04	8.04
Percentage Effects						
Post x One-Building Team	-16.06%	-21.35%	-14.45%	-17.58%	-14.15%	-16.89%
One-Building	14.46%	22.63%	22.38%	31.15%		
% One-Building Team	58.33	58.33	58.33	58.33	58.33	58.33
Local-Linear Time-Trends		✓		✓		✓
Preferreed Controls			✓	✓	✓	✓
All Controls					✓	✓
# Engineers	1,055	1,055	1,055	1,055	1,055	1,055
# Months	17	17	17	17	17	17
# Engineer-Months	9,304	9,304	9,304	9,304	9,304	9,304
R^2	0.02	0.02	0.36	0.36	0.50	0.50

Notes: This table tests the robustness of the results in Table 2 to the inclusion of local-linear time-trends on each side of the office closures for engineers on one- and multi-building teams. The odd columns repeat the results from Table 2 for reference. The even columns include local-linear time-trends that allow comments on each program to evolve differentially over time for engineers on one- and multi-building teams both before and after the offices closed for the pandemic. The preferred controls are those in Column 4 of Table 2 and all controls are those in Column 8. See Table 2's note for details on controls. Standard errors are clustered by engineering team. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

Table A.6: Proximity to Teammates and Online Feedback for Engineers who Work on Internal Tools

	Comments per Program							
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Post x One-Building Team	-1.52^{***}	-1.52^{***}	-1.58^{***}	-1.38^{***}	-1.39^{***}	-1.16^{**}	-1.78^{***}	-1.06^{**}
	(0.59)	(0.59)	(0.51)	(0.50)	(0.50)	(0.48)	(0.50)	(0.50)
One-Building Team	1.16	1.16	2.28^{***}	1.98^{***}	1.98^{***}			
	(0.74)	(0.74)	(0.65)	(0.59)	(0.59)			
Post	-1.36^{***}							
	(0.39)							
Pre-Mean, One-Building Team	8.27	8.27	8.27	8.27	8.27	8.27	8.27	8.27
Percentage Effects								
Post x One-Building Team	-18.3%	-18.3%	-19.1%	-16.7%	-16.8%	-14%	-21.5%	-12.8%
One-Building	14.1%	14.1%	27.6%	24%	24%			
% One-Building Team	58.3	58.3	58.3	58.3	58.3	58.3	58.3	58.3
Engineer Group x Post FE		✓	✓	✓	✓	✓	✓	✓
Program Scope Quartics			✓	✓	✓	✓	✓	✓
Months at Firm x Post FE				✓	✓	✓	✓	✓
Team Size x Post					✓	✓	✓	✓
Engineer FE						✓	✓	✓
Engineer Traits x Post FE							✓	✓
Main Building x Post FE								✓
# Teams	140	140	140	140	140	140	140	140
# Engineers	588	588	588	588	588	588	588	588
# Engineer-Months	5,630	5,630	5,630	5,630	5,630	5,630	5,630	5,630
R^2	0.02	0.02	0.28	0.36	0.36	0.49	0.50	0.50

Notes: This table replicates Table 2 but limits the sample to engineers who built internal tools for others in the firm. Since it was useful for these engineers to sit near the other engineers who used their tools, these engineers' teams were more likely to end up being split across the two buildings on the firm's main campus. Standard errors are clustered by team. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

Table A.7: Proximity to Teammates and Online Feedback for Engineers: Coded Comments

Panel (a): One- vs. Multi-Building Teams

	% Of Comments									
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
	Helpful		Implemented in Code		Actionable Feedback		Reasoning Explained		Rude	
One-Building x Post	2.09	−0.23	0.81	−0.01	3.43	1.06	−2.62	−3.16	−1.79	−1.91
	(3.38)	(2.51)	(4.41)	(4.48)	(4.04)	(3.83)	(3.50)	(3.84)	(2.56)	(2.64)
One-Building Team	0.79	1.32	2.81	3.58	0.86	1.14	3.85	3.80	1.30	1.05
	(2.43)	(1.83)	(2.64)	(2.67)	(2.49)	(2.43)	(2.50)	(2.71)	(1.88)	(1.92)
Dependent Mean	76.2%	87.0%	52.0%	70.4%	60.3%	68.4%	51.4%	58.4%	14.7%	15.2%

Panel (b): By Gender

	Helpful		Implemented in Code		Actionable Feedback		Reasoning Explained		Rude	
Female: One-Building x Post	2.27	0.45	−2.98	−6.62	0.65	−0.14	−6.59	−6.31	−4.06	−4.21
	(4.55)	(3.36)	(6.15)	(6.20)	(5.36)	(5.03)	(4.15)	(4.61)	(3.55)	(3.66)
Female: One-Building	1.08	1.74	4.82	7.58**	2.25	2.76	5.88*	6.63**	2.52	2.30
	(3.27)	(2.37)	(3.78)	(3.60)	(3.47)	(3.15)	(3.03)	(3.15)	(2.64)	(2.72)
Male: One-Building x Post	2.19	−1.07	4.01	5.32	6.18	2.13	0.85	−1.01	−0.01	−0.18
	(3.99)	(3.11)	(4.34)	(4.59)	(4.54)	(4.35)	(4.54)	(4.77)	(2.94)	(3.02)
Male: One-Building	0.90	1.72	1.59	1.00	−0.09	0.27	2.15	1.93	0.29	0.06
	(2.76)	(2.31)	(2.86)	(3.16)	(2.93)	(2.95)	(3.32)	(3.58)	(2.31)	(2.35)
Female: Dependent Mean	77.4%	88.7%	53.6%	72.1%	61.7%	69.8%	51.5%	58.7%	14.5%	15.0%
Male: Dependent Mean	74.9%	85.4%	50.4%	68.7%	58.9%	66.9%	51.4%	58.0%	14.8%	15.4%

Panel (c): By Tenure

	Helpful		Implemented in Code		Actionable Feedback		Reasoning Explained		Rude	
Junior (< 16mo): One-Building x Post	1.13	0.43	0.45	−1.65	4.68	2.67	−3.09	−3.57	−2.02	−1.94
	(3.69)	(2.82)	(4.79)	(4.95)	(4.56)	(4.38)	(4.12)	(4.65)	(2.90)	(2.96)
Junior (< 16mo): One-Building	1.45	1.37	2.59	3.60	−0.45	−0.55	2.99	3.12	0.90	0.54
	(2.67)	(2.02)	(2.98)	(3.14)	(2.92)	(2.66)	(2.80)	(2.98)	(2.00)	(2.04)
Senior (≥ 16mo): One-Building x Post	4.52	−1.07	0.95	2.26	−0.48	−3.89	−3.43	−3.66	−2.17	−2.70
	(4.93)	(3.87)	(5.93)	(6.04)	(5.50)	(5.75)	(4.98)	(5.35)	(4.26)	(4.43)
Senior (≥ 16mo): One-Building	−1.22	1.13	3.47	3.58	4.84	6.20	6.53	5.83	2.57	2.62
	(3.86)	(3.23)	(4.43)	(4.65)	(4.65)	(5.11)	(3.99)	(4.41)	(3.49)	(3.61)
Junior (< 16mo): Dependent Mean	77.5%	87.7%	54.2%	73.0%	62.9%	70.9%	53.3%	60.2%	13.8%	14.4%
Senior (≥ 16mo): Dependent Mean	74.0%	85.9%	48.5%	66.1%	56.0%	64.1%	48.4%	55.2%	16.0%	16.6%
Preferred Controls	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Drop Not Enough Info		✓		✓		✓		✓		✓
# Comments	5,377	4,708	5,377	3,974	5,377	4,741	5,377	4,740	5,377	5,174
# Teams	258	255	258	253	258	254	258	254	258	258

Notes: This table considers the random subset of comments that engineers evaluated on their helpfulness and content. Each column replicates our preferred specification in Column 4 of Table 2. The odd columns include all evaluated comments. The even columns exclude comments where the evaluator said that there was "not enough information" to determine the best response. For columns 1−2, the evaluators, were asked, "Would you find this comment helpful?" (Yes/No/Not enough info). For columns 3−4, "Do you think you would change your code because of this comment?" For columns 5−6, "Does this comment suggest actionable steps to change your code?" For columns 7−8, "Does this comment explain the reason for changing your code?" For columns 9−10, "Is the tone of this comment rude?" Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.8: Proximity to Teammates and Mentorship

Panel (a): Comments Received by Seniority

	Comments per Program					
	Received by Junior (< 16mo)			Received by Senior (≥ 16mo)		
	(1)	(2)	(3)	(4)	(5)	(6)
Post x One-Building Team	-1.40^{**}	-1.72^{***}	-2.01^{***}	-0.41	-0.38	0.25
	(0.65)	(0.60)	(0.63)	(0.45)	(0.42)	(0.67)
One-Building Team	1.27^{*}	2.61^{***}		0.13	0.58	
	(0.68)	(0.69)		(0.39)	(0.41)	
Post	-2.17^{***}			-0.04		
	(0.50)			(0.32)		
Pre-Mean, One-Building Team	9.56	9.56	9.56	4.99	4.99	4.99
Percentage Effects						
Post x One-Building Team	-14.7%	-18%	-21%	-8.2%	-7.7%	4.9%
One-Building Team	13.2%	27.3%		2.5%	11.7%	
Preferred Controls		✓			✓	
All Controls			✓			✓
# Engineer-Months	6,056	6,056	6,056	3,248	3,248	3,248

Panel (b): Comments Written by Seniority

	Comments per Program					
	Written by Junior (< 16mo)			Written by Senior (≥ 16mo)		
	(1)	(2)	(3)	(4)	(5)	(6)
Post x One-Building Team	-0.41^{*}	-0.25	-0.37	-0.95^{**}	-1.00^{***}	-0.84^{*}
	(0.24)	(0.26)	(0.32)	(0.38)	(0.36)	(0.45)
One-Building Team	0.15	0.39		1.07^{***}	1.47^{***}	
	(0.28)	(0.30)		(0.37)	(0.40)	
Post	-0.74^{***}			-0.42		
	(0.18)			(0.30)		
Pre-Mean, One-Building Team	2.94	2.94	2.94	3.73	3.73	3.73
Percentage Effects						
Post x One-Building Team	-13.9%	-8.6%	-12.7%	-25.4%	-26.8%	-22.4%
One-Building Team	5.1%	13.3%		28.7%	39.4%	
Preferred Controls		✓			✓	
All Controls			✓			✓
# Engineer-Months	9,304	9,304	9,304	9,304	9,304	9,304

Notes: This table investigates the relationship between sitting near teammates and (a) the feedback received by junior and senior engineers and (b) the feedback given by these engineers. The preferred controls reflect Column 4 of Table 2; all controls reflect Column 7. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.9: Externalities from a Distant Teammate on Online Feedback from Proximate Teammates

	# Comments			
	All Per Program (1)	From Proximate Teammates Per Review (2)	All Per Program (3)	From Proximate Teammates Per Review (4)
Post x On One-Building Team	-1.16^{***}	-0.54^{**}	-1.56^{***}	-0.71^{**}
	(0.43)	(0.26)	(0.46)	(0.28)
On One-Building Team	1.80^{***}	0.71^{***}		
	(0.49)	(0.28)		
Pre-Mean, One-Building Team	8.04	4.17	8.04	4.17
Percentage Effects				
Post x On One-Building Team	-14.45%	-12.98%	-19.36%	-17.1%
On One-Building Team	22.38%	17.07%		
Avg. on Multi-Building Teams				
# Teammate Commenters	1.71	1.71	1.71	1.71
% From Proximate Teammates	39.39	39.39	39.39	39.39
# Proximate Teammate Commenters	0.67	0.67	0.67	0.67
Back-of-the-envelope Calculations				
% Initial Gap Explained		26.59%		
% Differential Change Explained		31.33%		30.8%
Controls	Preferred	Preferred	All	All
# Engineers	1,055	934	1,055	934
# Engineer-Months	9,304	7,174	9,304	7,174
R^2	0.36	0.24	0.50	0.46

Notes: This table investigates whether having a teammate in a different building impacts the online feedback an engineer receives from her proximate (same-building) teammates. The odd columns consider all comments on each program. The even columns consider the average length of reviews from proximate teammates, conditional on them leaving reviews. The first two columns include the preferred controls. The next two columns include all controls. The back-of-the-envelope calculations consider how much feedback from proximate teammates can explain overall effects on comments in the preceding column, based on the share of comments that come from proximate teammates. Each column estimates Equation 2. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.10: Difference-in-Differences Around New Hires in a Different Building From Teammates vs. Other Hires Before COVID-19

	Comments per Review from Same-Building Teammate		
Post Hire x One- to Multi-Building Team	-1.483^{**}	-1.710^{*}	-1.332
	(0.640)	(1.017)	(1.062)
Post Hire	-0.302	-0.068	0.010
	(0.987)	(1.151)	(1.008)
Bandwidth = 6 weeks	✓	✓	✓
Pre-Period Mean for Treated	4.329	4.329	4.329
Engineer x Event FE	✓	✓	✓
Engineer x Commenter x Event FE		✓	✓
Program Content			✓
# Teams	126	126	126
# Treated Teams	16	16	16
# Engineers	400	400	400
# Treated Engineers	46	46	46
# Engineer-Commenter Pairs	1159	1159	1159
# Treated Engineer-Commenter Pairs	142	142	142
Observations	4,017	4,017	4,017
R^2	0.236	0.407	0.552

Notes: This table compares the change in comments per review in teams where a new hire converts the team from a one-building team to a multi-building team relative to teams where a new hire does not change whether they are a one- or a multi-building team. Each observation is the comments that a particular commenter left on a coder's program. The analysis compares the change in the length of the peer-reviews in the commenter-coder pair around the two types of new hires as in Equation 4. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.11: Proximity to Teammates and Engineer Output for Engineers who Build Internal Tools

	(1)	(2)	(3)	(4)	(5)	(6)	(7)
	Panel (a): Programs per Month						
Post x One-Building Team	0.47***	0.47***	0.44***	0.44***	0.39***	0.38***	0.44***
	(0.14)	(0.14)	(0.13)	(0.13)	(0.13)	(0.13)	(0.15)
One-Building Team	−0.42**	−0.42**	−0.39**	−0.41**			
	(0.19)	(0.19)	(0.19)	(0.19)			
Pre-Mean, One-Building Team	1.82	1.82	1.82	1.82	1.82	1.82	1.82
Post x One-Building Team as %	25.8%	25.8%	24.2%	24.2%	21.3%	20.9%	24%
One-Building Team as %	-23.2%	-23.2%	-21.6%	-22.7%			
R^2	0.01	0.01	0.04	0.05	0.42	0.43	0.43
	Panel (b): Lines Added per Month						
Post x One-Building Team	100**	100**	103**	105**	97**	86**	153***
	(43)	(43)	(44)	(42)	(42)	(43)	(46)
One-Building Team	−197***	−197***	−202***	−209***			
	(52)	(52)	(53)	(53)			
Pre-Mean, One-Building Team	332	332	332	332	332	332	332
Post x One-Building Team as %	30.1%	30.1%	30.8%	31.6%	29.2%	26%	46.2%
One-Building Team as %	-59.4%	-59.4%	-60.9%	-63%			
R^2	0.02	0.02	0.05	0.05	0.34	0.35	0.35
	Panel (c): Files Changed per Month						
Post x One-Building Team	1.50	1.50	1.41	1.50	1.10	0.90	2.11*
	(1.05)	(1.05)	(1.04)	(1.03)	(1.02)	(1.03)	(1.11)
One-Building Team	−4.12***	−4.12***	−4.11***	−4.33***			
	(1.25)	(1.25)	(1.27)	(1.26)			
Pre-Mean, One-Building Team	9.03	9.03	9.03	9.03	9.03	9.03	9.03
Post x One-Building Team as %	16.6%	16.6%	15.7%	16.6%	12.2%	10%	23.4%
One-Building Team as %	-45.7%	-45.7%	-45.5%	-47.9%			
R^2	0.01	0.01	0.04	0.04	0.32	0.33	0.33
Engineer Group x Post FE		✓	✓	✓	✓	✓	✓
Months at Firm x Post FE			✓	✓	✓	✓	✓
Team Size x Post				✓	✓	✓	✓
Engineer FE					✓	✓	✓
Engineer Traits x Post FE						✓	✓
Main Building x Post FE							✓
# Teams	304	304	304	304	304	304	304
# Engineers	1,055	1,055	1,055	1,055	1,055	1,055	1,055
# Engineer-Months	9,304	9,304	9,304	9,304	9,304	9,304	9,304

Notes: This table replicates Table 4 but limits the sample to engineers who built internal tools for others in the firm. Since it was useful for these engineers to sit near the other engineers who used their tools, these engineers' teams were more likely to end up being split across the two buildings on the firm's main campus. Standard errors are clustered by engineering team. *p<0.1; **p<0.05; ***p<0.01.

Table A.12: Proximity to Teammates and Engineer Output by Seniority

	Monthly Programming Output					
	Programs		Lines of Code		Files Changed	
	(1)	(2)	(3)	(4)	(5)	(6)
Junior x Post x One-Building Team	0.35**	0.26*	92.29**	96.58**	0.93	0.82
	(0.14)	(0.14)	(46.23)	(45.75)	(1.16)	(1.11)
Junior x One-Building Team	−0.13		−160.40***		−2.86**	
	(0.18)		(53.49)		(1.28)	
Senior x Post x One-Building Team	0.49***	0.47**	90.34*	157.40**	2.11	2.88**
	(0.18)	(0.20)	(53.16)	(61.63)	(1.29)	(1.40)
Senior x One-Building Team	−0.64***		−176.00***		−4.54***	
	(0.21)		(57.58)		(1.40)	
Junior Pre-Mean, One-Building Team	1.75	1.75	339.27	339.27	9.62	9.62
Senior Pre-Mean, One-Building Team	1.65	1.65	278.17	278.17	8.32	8.32
Percentage Effects						
Junior x Post x One-Building Team	19.9%	14.8%	27.2%	28.5%	9.7%	8.5%
Junior x One-Building Team	-7.3%		-47.3%		-29.7%	
Senior x Post x One-Building Team	29.5%	28.4%	32.5%	56.6%	25.3%	34.6%
Senior x One-Building Team	-38.5%		-63.3%		-54.6%	
Preferred Controls	✓	✓	✓	✓	✓	✓
All Controls		✓		✓		✓
# Teams	304	304	304	304	304	304
# Engineers	1,055	1,055	1,055	1,055	1,055	1,055
# Engineer-Months	16,058	16,058	16,058	16,058	16,058	16,058

Notes: This table investigates the relationship between sitting near teammates and monthly programming output, separately for junior and senior engineers. Junior engineers are defined as those with less than 16 months of experience at the firm before the office closures and senior engineers as those with at least 16 months at the firm (the median tenure). The preferred controls and full set of controls are described in Section III. Standard errors are clustered by engineering team. *p$<$0.1; **p$<$0.05; ***p$<$0.01.

Table A.13: Gendered Impacts of Proximity

	Panel (a): Difference-in-Differences in Feedback Per Program				
	Comments	# Commenters	Commenter Initial Comments	Author Follow-up Questions	Commenter Follow-up Comments
	(1)	(2)	(3)	(4)	(5)
Female: Post x One-Building Team	-2.83^{***}	-0.31^{***}	-1.02^{**}	-0.17^{*}	-1.81^{***}
	(0.81)	(0.09)	(0.52)	(0.10)	(0.70)
Female: One-Building Team	3.84^{***}	0.37^{***}	1.15^{**}	0.23^{**}	2.69^{***}
	(0.77)	(0.09)	(0.48)	(0.09)	(0.66)
Male: Post x One-Building Team	-0.83^{*}	-0.06	-0.37	-0.14^{***}	-0.46
	(0.45)	(0.05)	(0.27)	(0.05)	(0.34)
Male: One-Building Team	1.38^{***}	0.17^{***}	0.68^{**}	0.09^{*}	0.70^{*}
	(0.52)	(0.06)	(0.27)	(0.05)	(0.40)
Female Pre-Mean, One-Building Team	9.56	1.87	5.38	0.32	4.18
Male Pre-Mean, One-Building Team	7.67	1.75	4.79	0.22	2.88
Percentage Effects					
Female: Post x One-Building Team	-29.6%	-16.4%	-18.9%	-54.6%	-43.4%
Female: One-Building	40.1%	19.8%	21.3%	72.6%	64.3%
Male: Post x One-Building Team	-10.8%	-3.5%	-7.8%	-61.3%	-15.8%
Male: One-Building	18%	9.7%	14.2%	41%	24.3%

	Panel (b): Triple Difference in Feedback Per Program				
	Comments	# Commenters	Commenter Initial Comments	Author Follow-up Questions	Commenter Follow-up Comments
	(1)	(2)	(3)	(4)	(5)
Female x Post x One-Building Team	-2.01^{**}	-0.25^{***}	-0.65	-0.04	-1.36^{**}
	(0.82)	(0.08)	(0.50)	(0.09)	(0.67)
Female x One-Building Team	2.46^{***}	0.20^{***}	0.47	0.14^{*}	1.99^{***}
	(0.76)	(0.08)	(0.47)	(0.08)	(0.61)
Percentage Effects					
Female x Post x One-Building Team	-21%	-13.1%	-12%	-12.2%	-32.5%
Female x One-Building	25.7%	10.7%	8.7%	44.2%	47.6%
# Teams	304	304	304	304	304
# Engineers	1,055	1,055	1,055	1,055	1,055
# Engineer-Months	9,304	9,304	9,304	9,304	9,304

Notes: This table investigates the gendered impacts of proximity to teammates. Panel (a) shows difference-in-differences designs, conditional on our preferred set of controls (as in Table 3), for male and female programmers. This allows the difference-in-differences coefficients to be different by gender, while allowing the controls to be estimated jointly. Panel (b) shows the triple difference design, testing the difference in the estimated effects for female and male engineers. The sample includes engineers who submit programs to the firm's main code-base in the month and whose teams are all in the firm's main campus. Standard errors are clustered by team. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

Table A.14: Proximity, Gender and Dimensions of Online Feedback

Panel (a): Feedback Length, Delay, and Mentions of Other Online Conversations

	Comments per Program	Total Characters	Hours to Comment	% Other Online Convo
	(1)	(2)	(3)	(4)
Female: Post x One-Building Team	-2.83^{***}	-338.20^{***}	1.93	0.90
	(0.81)	(110.70)	(1.29)	(1.63)
Female: One-Building Team	3.84^{***}	469.90^{***}	-2.60^{***}	1.59
	(0.77)	(102.10)	(0.96)	(1.31)
Male: Post x One-Building Team	-0.83^{*}	-98.23	0.83	-1.66^{*}
	(0.45)	(62.61)	(0.62)	(0.92)
Male: One-Building Team	1.38^{***}	147.10^{**}	-1.20^{**}	2.55^{***}
	(0.52)	(66.25)	(0.61)	(0.95)
Pre-Mean, Female, One-Building Team	9.56	1048.15	14.78	3.56
Pre-Mean, Male, One-Building Team	7.67	781.09	16.33	4.18
Percentage Effects				
Female: Post x One-Building Team	-29.6%	-32.3%	13.1%	25.4%
Female: One-Building Team	40.1%	44.8%	-17.6%	44.7%
Male: Post x One-Building Team	-10.8%	-12.6%	5.1%	-39.7%
Male: One-Building Team	18%	18.8%	-7.4%	61%

Panel (b): Intensive and Extensive Margins

	Comments per Program	Total Characters	Hours to Comment	% Other Online Convo
	(1)	(2)	(3)	(4)
Female x Post x One-Building Team	-2.01^{**}	-240.00^{**}	1.11	2.56
	(0.82)	(119.10)	(1.32)	(1.80)
Female x One-Building Team	2.46^{***}	322.80^{***}	-1.39	-0.96
	(0.76)	(104.90)	(0.94)	(1.37)
Percentage Effects				
Female x Post x One-Building Team	-21%	-22.9%	7.5%	71.9%
Female x One-Building	25.7%	30.8%	-9.4%	-26.8%
# Teams	304	304	304	304
# Engineers	1,055	1,055	1,055	1,055
# Engineer-Months	9,304	9,304	9,304	9,304

Notes: This table considers alternative metrics of the extent and timeliness of feedback by gender. Panel (a) shows difference-in-differences designs, conditional on our preferred set of controls (as in Table 3), for male and female programmers. This allows the difference-in-differences coefficients to be different by gender, while allowing the controls to be estimated jointly. Panel (b) shows the triple difference design, testing the difference in the estimated effects for female and male engineers. The sample includes engineers who submit programs to the firm's main code-base in the month and whose teams are all in the firm's main campus. Standard errors are clustered by team. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

Table A.15: Gendered Impacts of Proximity Conditional on Tenure & Age

	Comments per Program					
	(1)	(2)	(3)	(4)	(5)	(6)
Female x Post x One-Building Team	-2.01^{**}	-1.98^{**}	-1.83^{**}	-1.65^{*}	-1.83^{*}	-1.68^{*}
	(0.82)	(0.81)	(0.86)	(0.88)	(0.97)	(1.01)
Female x One-Building Team	2.46^{***}	2.42^{***}	2.47^{***}	2.32^{***}	2.42^{***}	2.28^{**}
	(0.76)	(0.75)	(0.80)	(0.82)	(0.93)	(0.97)
Female Pre-Mean, One-Building Team	9.56	9.56	9.56	9.56	9.56	9.56
Male Pre-Mean, One-Building Team	7.67	7.67	7.67	7.67	7.67	7.67
Percentage Effects						
Female x Post x One-Building Team	-21%	-20.7%	-19.2%	-17.3%	-19.1%	-17.6%
Female x One-Building	25.7%	25.3%	25.8%	24.3%	25.3%	23.8%
Diff-in-Diff Interaction						
Junior (Months at Firm < 16mo)		✓	✓	✓	✓	✓
Months at Firm FE			✓	✓	✓	✓
Age < 30				✓	✓	✓
Age FE					✓	✓
Age < 30 x Junior						✓
# Teams	304	304	304	304	304	304
# Engineers	1,055	1,055	1,055	1,055	1,055	1,055
# Engineer-Months	9,304	9,304	9,304	9,304	9,304	9,304

Notes: This table investigates the gendered impacts of proximity to teammates while allowing for differential effects of proximity by gender and age. Column 1 repeats Column 1 of Table A.13(b) for reference. Column 2 allows for the effect of being on a one-building team to vary for junior versus senior engineers. Column 3 allows this interaction to vary by the precise number of months that the worker has been at the firm. Column 4 further allows the effect of proximity to differ for young engineers under 30. Column 5 allows for the effect of proximity to differ for engineers of each age. Column 6 allows for an interaction between being junior and young. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.16: Gendered Impacts of Proximity on Mentorship and Output

	Panel (a): Difference-in-Differences in Mentorship & Output			
	Comments per Review	Programs	Lines of Code	Files Changed
	(1)	(2)	(3)	(4)
Female: Post x One-Building Team	-1.31^{**}	0.62^{***}	134.70^{**}	3.77^{**}
	(0.60)	(0.22)	(57.73)	(1.51)
Female: One-Building Team	1.13^{**}	-0.39^{*}	-128.50^{*}	-3.89^{**}
	(0.57)	(0.24)	(68.82)	(1.81)
Male: Post x One-Building Team	-0.20	0.37^{***}	82.21^{*}	0.88
	(0.22)	(0.14)	(43.28)	(1.06)
Male: One-Building Team	0.32	-0.32^{*}	-176.00^{***}	-3.50^{***}
	(0.27)	(0.17)	(48.84)	(1.14)
Female Pre-Mean, One-Building Team	4.15	1.38	295.56	7.78
Male Pre-Mean, One-Building Team	3.96	1.8	322.75	9.51
Percentage Effects				
Female: Post x One-Building Team	-31.5%	44.7%	45.6%	48.5%
Female: One-Building	27.2%	-28.4%	-43.5%	-49.9%
Male: Post x One-Building Team	-5%	20.4%	25.5%	9.3%
Male: One-Building	8.1%	-17.9%	-54.5%	-36.8%

	Panel (b): Triple Difference in Mentorship & Output			
	Comments per Review	Programs	Lines of Code	Files Changed
	(1)	(2)	(3)	(4)
Female x Post x One-Building Team	-1.11^{*}	0.25	52.48	2.89^{*}
	(0.62)	(0.24)	(62.84)	(1.64)
Female x One-Building Team	0.81	-0.07	47.52	-0.39
	(0.57)	(0.24)	(74.43)	(1.86)
Percentage Effects				
Female x Post x One-Building Team	-26.7%	18.2%	17.8%	37.2%
Female x One-Building	19.5%	-5%	16.1%	-5%
# Teams	304	304	304	304
# Engineers	1,055	1,055	1,055	1,055
# Engineer-Months	16,058	16,058	16,058	16,058

Notes: This table investigates the gendered impacts of proximity to teammates on mentorship given and programming output. Panel (a) shows difference-in-differences designs, conditional on our preferred set of controls for engineering group and tenure at the firm, for male and female programmers. This allows the difference-in-differences coefficients to be different by gender, while allowing the controls to be estimated jointly. Panel (b) shows the triple difference design, testing the difference in the estimated effects for female and male engineers. The sample includes engineers who ever submit programs to the firm's main code-base and whose teams are all in the firm's main campus. Standard errors are clustered by team. $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$.

Table A.17: Effect of Proximity to Teammates on Pay Raises for Engineers Who Built Internal Tools

	% Pay Raise					
	Offices Open			Offices Closed		
	(1)	(2)	(3)	(4)	(5)	(6)
One-Building Team	-3.78^{**}			4.72		
	(1.62)			(4.11)		
Junior (< 16mo) x One-Building Team		-5.42^{**}			4.90	
		(2.38)			(4.61)	
Senior (≥ 16mo) x One-Building Team		-1.59			1.73	
		(5.70)			(7.12)	
Female x One-Building Team			-15.68^{**}			9.17
			(6.21)			(11.05)
Male x One-Building Team			-2.97			4.04
			(2.36)			(3.96)
Dependent Mean	13.53	13.53	13.53	38.72	38.72	38.72
Junior (< 16mo) Mean	17.21	17.21	17.21	17.21	17.21	17.21
Senior (≥ 16mo) Mean	24.46	24.46	24.46	42.97	42.97	42.97
Female Mean	21.67	21.67	21.67	36.36	36.36	36.36
Male Mean	19.19	19.19	19.19	39.1	39.1	39.1
Percentage Effect						
One-Building Team	-27.9%			12.2%		
Junior (< 16mo) x One-Building Team		-31.5%			13.3%	
Senior (≥ 16mo) x One-Building Team		-6.5%			4%	
Female x One-Building Team			-72.4%			25.2%
Male x One-Building Team			-15.5%			10.3%
Preferred Controls	✓	✓	✓	✓	✓	✓
# Teams	262	262	262	256	256	256
# Engineers	801	801	801	720	720	720
# Engineer-Review	1,988	1,988	1,988	1,851	1,851	1,851

Notes: This table replicates Table 5 but limits the sample to engineers who worked on internal tools used by other workers at the firm. Since it was useful for these engineers to sit near the other engineers who used their tools, these engineers' teams were more likely to end up being split across the two buildings on the firm's main campus. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.18: Effect of Proximity to Teammates on Salary Increases

	Offices Open			Offices Closed		
	(1)	(2)	(3)	(4)	(5)	(6)
	(a): Salary Increase (in $s)					
One-Building Team	−238			239		
	(221)			(303)		
Junior (< 16mo) x One-Building Team		−142			228	
		(316)			(341)	
Senior (≥ 16mo) x One-Building Team		−453			361	
		(661)			(495)	
Female x One-Building Team			−923			924
			(764)			(798)
Male x One-Building Team			−86			111
			(313)			(324)
Dependent Mean	1,626	1,626	1,626	2,867	2,867	2,867
(Junior, Senior) Means	(1.9K,3.2K)	(1.9K,3.2K)	(1.9K,3.2K)	(2.6K,3.4K)	(2.6K,3.4K)	(2.6K,3.4K)
(Female, Male) Means	(2.8K,2.2K)	(2.8K,2.2K)	(2.8K,2.2K)	(3.4K,2.8K)	(3.4K,2.8K)	(3.4K,2.8K)
Percentage Effect						
One-Building Team	-14.6%			8.4%		
Junior (< 16mo) x One-Building Team		-7.5%			8.9%	
Senior (≥ 16mo) x One-Building Team		-14.3%			10.6%	
Female x One-Building Team			-32.5%			27.3%
Male x One-Building Team			-3.9%			4%
	(b): Inverse Hyperbolic Sine of Salary Increase					
One-Building Team	-0.32^{**}			0.36		
	(0.15)			(0.31)		
Junior (< 16mo) x One-Building Team		-0.47^{**}			0.62^{*}	
		(0.21)			(0.34)	
Senior (≥ 16mo) x One-Building Team		−0.08			−0.09	
		(0.38)			(0.45)	
Female x One-Building Team			−0.69			1.04
			(0.47)			(0.65)
Male x One-Building Team			−0.28			0.23
			(0.20)			(0.32)
Percentage Effect						
One-Building Team	-24.5%			9.7%		
Junior (< 16mo) x One-Building Team		-28.9%			17%	
Senior (≥ 16mo) x One-Building Team		-3.4%			-2.4%	
Female x One-Building Team			-34.5%			26.8%
Male x One-Building Team			-15.1%			6.2%
Preferred Controls	✓	✓	✓	✓	✓	✓
# Teams	262	262	262	256	256	256
# Engineers	801	801	801	720	720	720
# Engineer-Review	1,988	1,988	1,988	1,851	1,851	1,851

Notes: This table replicates the analysis of pay raises in Table 5 but considers the absolute changes in salaries in (a) and the inverse hyperbolic sine of these changes in (b). Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.19: Effect of Proximity to Teammates on Firings

	% Fired					
	Offices Open			Offices Closed		
	(1)	(2)	(3)	(4)	(5)	(6)
One-Building Team	−0.04			−0.18		
	(0.09)			(0.12)		
Junior (< 16mo) x One-Building Team		−0.12			−0.17	
		(0.12)			(0.13)	
Senior (≥ 16mo) x One-Building Team		0.11			−0.21	
		(0.11)			(0.29)	
Female x One-Building Team			0.01			−0.32
			(0.06)			(0.34)
Male x One-Building Team			−0.06			−0.15
			(0.11)			(0.13)
Dependent Mean	0.07	0.07	0.07	0.18	0.18	0.18
Junior (< 16mo) Mean	0.08	0.08	0.08	0.08	0.08	0.08
Senior (≥ 16mo) Mean	0.05	0.05	0.05	0.24	0.24	0.24
Female Mean	0	0	0	0.3	0.3	0.3
Male Mean	0.08	0.08	0.08	1.83	1.83	1.83
Percentage Effect						
One-Building Team	-59.2%			-97.2%		
Junior (< 16mo) x One-Building Team		-150.2%			-101.9%	
Senior (≥ 16mo) x One-Building Team		213.4%			-88.4%	
Female x One-Building Team			NA			-106.7%
Male x One-Building Team			-66%			-8%
Preferred Controls	✓	✓	✓	✓	✓	✓
# Teams	303	303	303	297	297	297
# Engineers	1,055	1,055	1,055	994	994	994
# Engineer-Month	6,812	6,812	6,812	9,288	9,288	9,288

Notes: This table investigates how the likelihood of being fired differs for engineers on one-building teams while the offices were open (Columns 1−3) and after the offices closed (Columns 4−6). Each column includes our preferred, time-varying controls for engineering type and firm tenure. Each observation is an engineer-month. The sample includes engineers who worked on the main code-base, whose teammates were all in the main campus, and who were hired before the start of our sample. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.

Table A.20: Effect of Proximity to Teammates on Quits to Higher and Lower Paying Jobs

	% Quit								
	All	Higher Pay	Lower Pay	All	Higher Pay	Lower Pay	All	Higher Pay	Lower Pay
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Post x One-Building Team	0.67**	0.43*	0.24						
	(0.29)	(0.24)	(0.17)						
One-Building Team	−0.22	−0.13	−0.09						
	(0.20)	(0.15)	(0.12)						
Junior (< 16mo) x Post x One-Building				0.80**	0.58**	0.22			
				(0.35)	(0.29)	(0.22)			
Junior (< 16mo) x One-Building				−0.24	−0.11	−0.13			
				(0.22)	(0.16)	(0.17)			
Senior (≥ 16mo) x Post x One-Building				0.41	0.11	0.30			
				(0.50)	(0.46)	(0.19)			
Senior (≥ 16mo) x One-Building				−0.20	−0.16	−0.05			
				(0.34)	(0.34)	(0.06)			
Female x Post x One-Building							2.46***	1.97**	0.50*
							(0.81)	(0.77)	(0.30)
Female x One-Building							−0.53	−0.59	0.06
							(0.58)	(0.57)	(0.13)
Male x Post x One-Building							0.30	0.11	0.19
							(0.32)	(0.26)	(0.19)
Male x One-Building							−0.15	−0.03	−0.12
							(0.21)	(0.14)	(0.14)
Dependent Mean	0.70%	0.55%	0.15%	0.70%	0.55%	0.15%	0.70%	0.55%	0.15%
Preferred Controls	✓	✓	✓	✓	✓	✓	✓	✓	✓
# Engineer-Months	13,169	13,169	13,169	13,169	13,169	13,169	13,169	13,169	13,169
# Engineers	849	849	849	849	849	849	849	849	849
# Engineer Teams	276	276	276	276	276	276	276	276	276

Notes: This table investigates how the likelihood of a quit for a higher or lower paying job differs for engineers on one- and multi-building teams around the office closures. We categorize a worker as quitting for higher pay if they quit for a position with higher average compensation (including base salary, bonuses, and stocks) as measured by Glassdoor data. We compare those who quit for higher pay to workers who did not quit or quit for a position with lower average compensation. We exclude workers who quit but for whom we do not know the destination compensation. The sample includes engineers who worked on the main code-base, whose teammates were all in the main campus, and who were hired before the start of our sample. Each column includes our preferred, time-varying controls for engineering type and firm tenure. Standard errors are clustered by team. *p<0.1; **p<0.05; ***p<0.01.