



# Vom Back-Office ins Home Office

Kollaboratives Remote Arbeiten für Entwickler

# Über mich

Birgit Kratz

Freiberufliche Softwareentwicklerin

✉ [mail@birgitkratz.de](mailto:mail@birgitkratz.de)

🐦 @bikratz



Co-Organisatorin der  
Softwerkskammern in Düsseldorf und Köln

Icons made by [Freepik](#) from [www.flaticon.com](#)

# Ein Jahr Home-Office

## Ein Erfahrungs- und Erkenntnisbericht

# Worüber will ich reden

Wie alles begann

Was man braucht

Tools

Zur Kommunikation

Zum gemeinsamen Programmieren

Zur Agilität

Abseits aller Hardware und Software

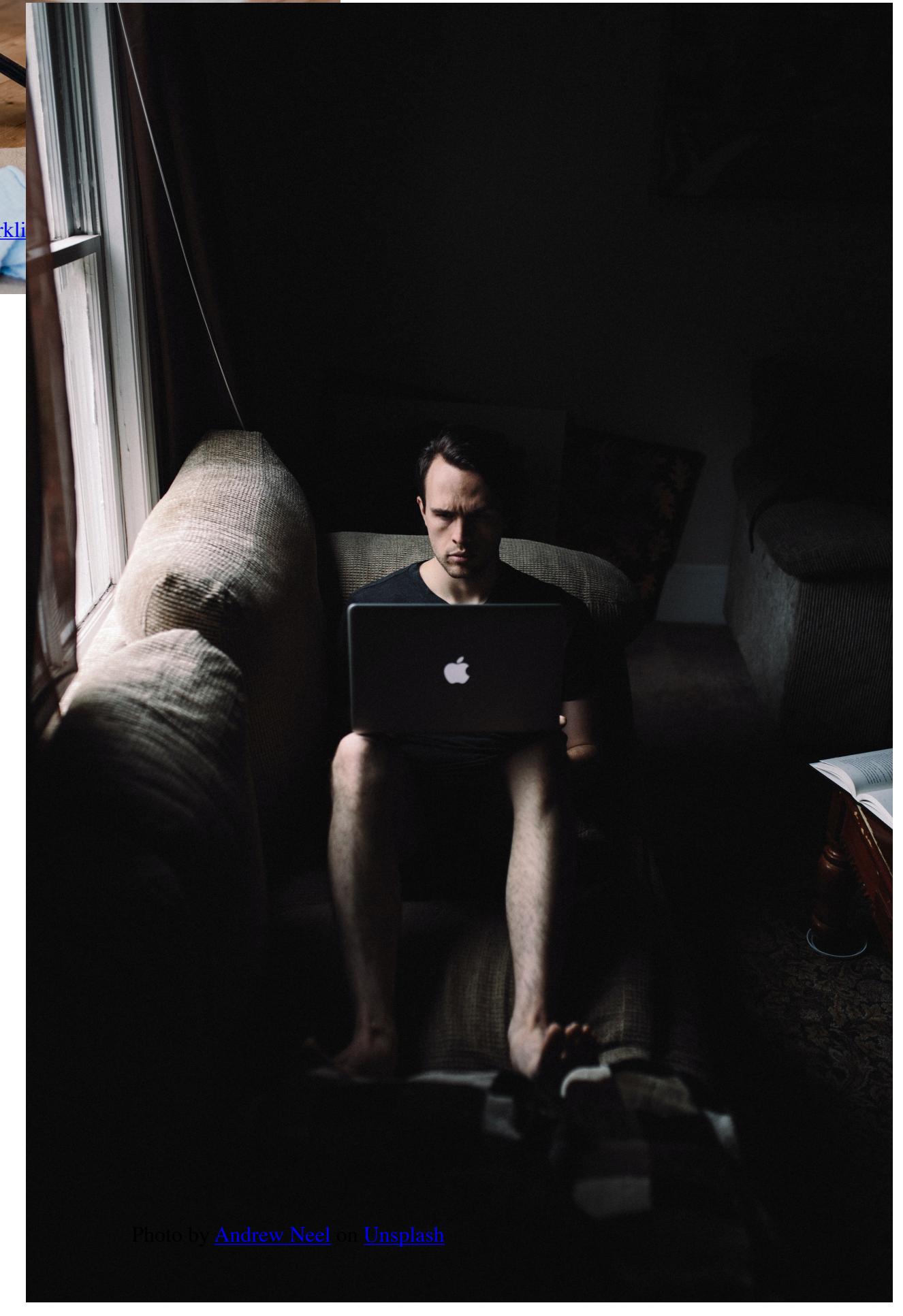
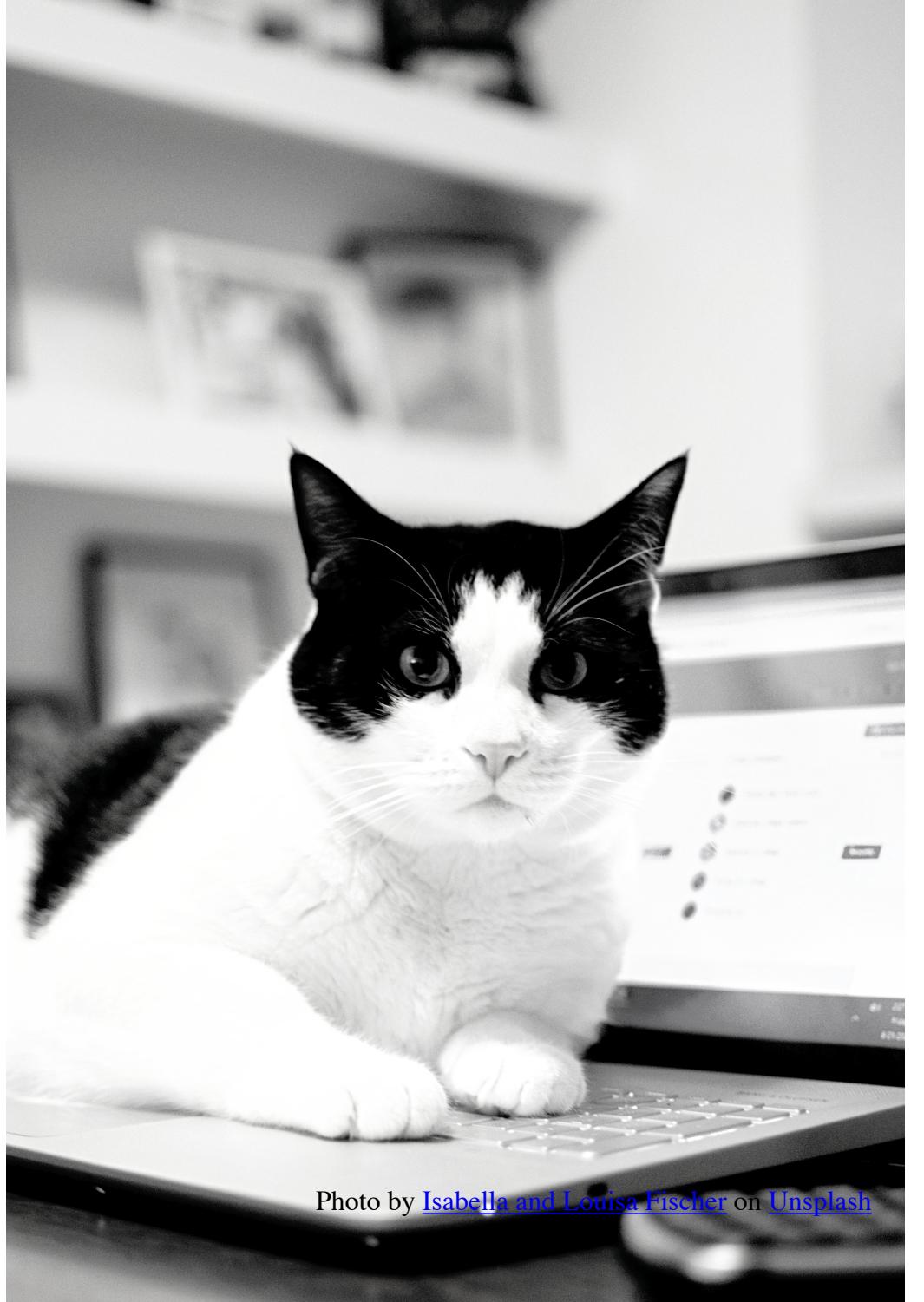
Ausblick

# Frage

Was hast du am **16. März 2020** gemacht?



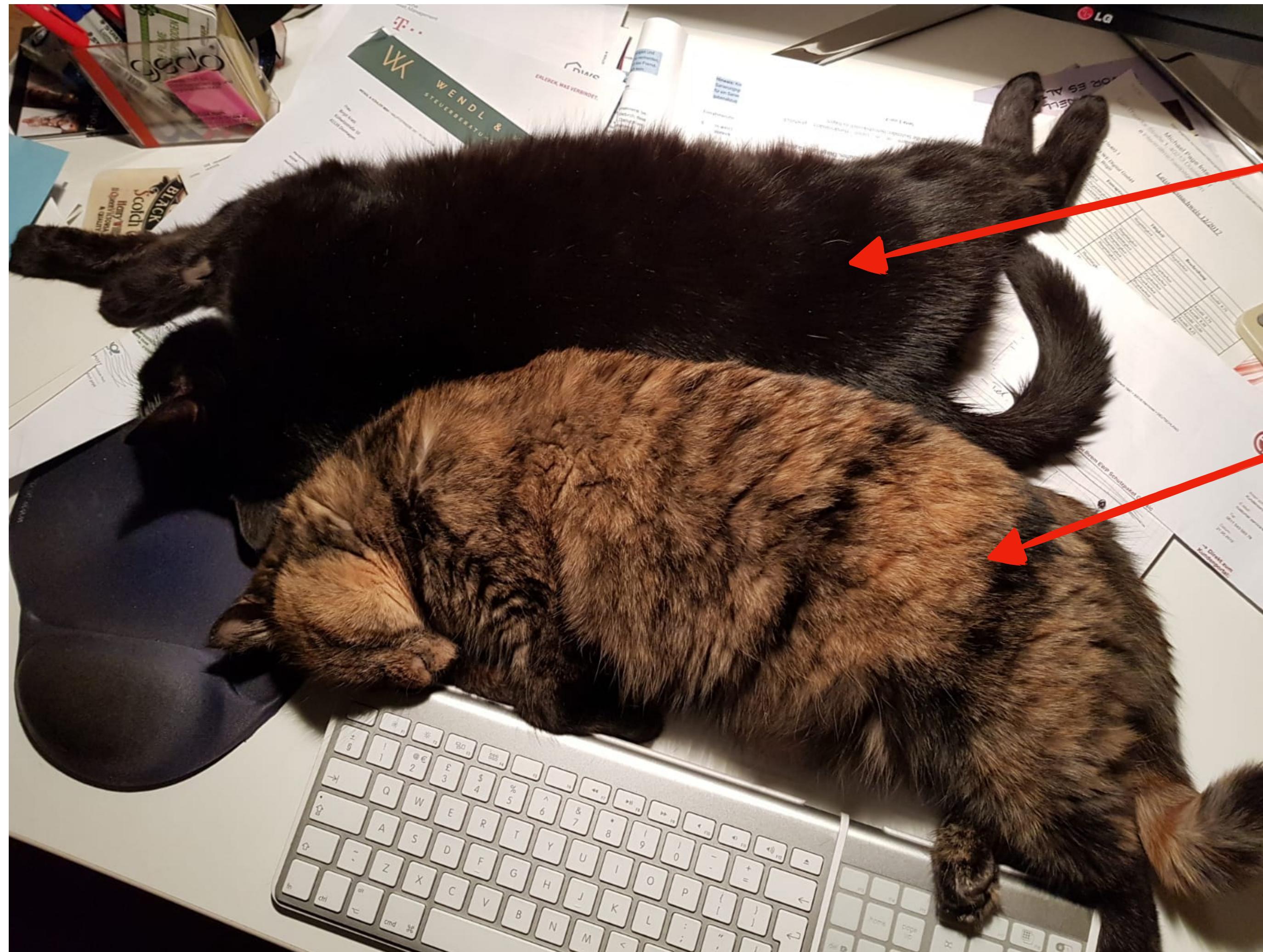
Morgens auf dem Weg ins  
Home Office 😊



# Mein Büro



Und manchmal sieht mein Schreibtisch so aus



Baghira

Peppels

# Wichtig für die HomeOffice Ausstattung

Schreibtisch

Stuhl (oder Hocker)

Bildschirm

Tastatur

Webcam

Mikrofon / Lautsprecher / Headset (noise cancelling)

Internetanbindung

Licht

Umgebungsgeräusche

Walking Pad (vielleicht demnächst, ich überlege noch)

# Chatten und Meeting

Slack

Teams

Zoom

Google Hangout

WebEx

Jitsi

Pragli

# Pragli (<https://pragli.com/>)

als App oder im Browser

in Anlehnung an die Struktur im Büro

man kann Büros, Meetingräume, Kaffeeküchen, etc. einrichten

per Klick auf einen dieser Räume wechselt man in den Raum.

man kann auf einen Blick sehen, wer sich in welchem Raum befindet

wenn man einen Raum betritt, ist per Default das Mikro an

man kann Räume locken, so dass ein Zutritt nur nach Anklopfen und Öffnen der Tür "von innen" möglich ist

Gedacht, um sich online in einem Büro zu treffen, Kamera an, Ton an und man kann einfach leise vor sich hin arbeiten und ab und zu mal was sagen (so wie es auch im Office oft war) woraus sich dann für eine Weile ein Gespräch entwickeln kann. Man kann schnell mal eine Frage in den Raum werfen. Man sitzt quasi gemeinsam in einem "Raum" und arbeitet, genauso wie im Office auch.

# Screensharing

Any Screen Sharing Tool (z.B. Slack Anruf mit Screen Sharing)

Vorteil:

- Kein Server involviert, über den Code geteilt wird, da reines Screensharing

Nachteil:

- Latenz
- Nur mündliche Interaktion möglich (Linux), manchmal auch Marker-Stift vorhanden

# IntelliJ Code With Me

Vorteil:

- funktioniert für alle IntelliJ IDEs (IDEA, PhpStorm, WebStorm, GoLand)
- Verschiedene Sharing-Permissions (Read-Only, Edit files, Full access, Custom)
- Integrierter Voice Call und Video Call
- Pairing Partner braucht auf seinem Rechner kein installiertes IntelliJ Produkt, das Sharing funktioniert über einen Client, der per Shell-Befehl heruntergeladen und installiert wird
- Immer mehr Tool-Windows werden in diesen Client integriert

Nachteil:

- Wenn kein On-Site Server vorhanden ist, geht die Kommunikation über Jet-Brains Server



**birgitkratz has invited you to join their project  
swagger-project in IntelliJ IDEA.**

Copy the following text and run it in the terminal.

```
/bin/bash -c "$(curl -fsSL 'https://code-with-me.jetbrains.com/zc9ou78oInJAKTcc1LoRkQ/cwm-client-launcher.sh')"
```

Alternatively you can launch IntelliJ IDEA version 2020.2 or newer with Code With Me plugin.

You can download the latest version of IntelliJ IDEA [here](#).

[About Code With Me](#)

# IntelliJ Code With Me

The screenshot displays two instances of the IntelliJ IDEA IDE running side-by-side, illustrating the 'Code With Me' feature. Both instances show a Java project structure with Maven configurations. The left instance has the focus on the 'pom.xml' file for the 'day17-java' module, while the right instance has the focus on the same file for the 'aoc-2020' module. The interface includes toolbars, menus, and various code navigation and analysis tools. At the bottom, a video call interface shows two users, indicating a live collaboration session.

# IntelliJ Code With Me

Hintergrundwissen

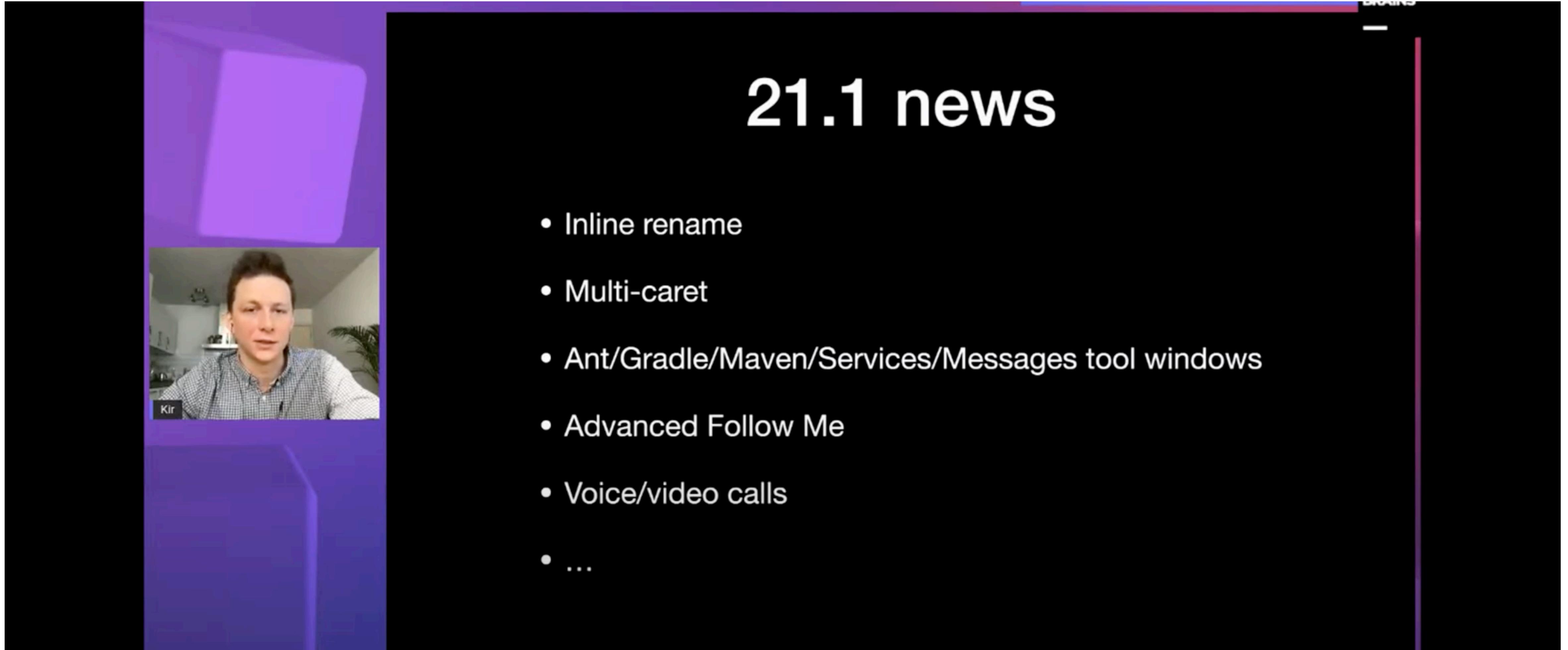
Youtube:

Code With Me - Behind the Scenes

<https://youtu.be/SrPqIxsfxF8>

IntelliJ IDEA Conf 2021: <https://youtu.be/akrPpWAZzQk>

# IntelliJ Code With Me



21.1 news

- Inline rename
- Multi-caret
- Ant/Gradle/Maven/Services/Messages tool windows
- Advanced Follow Me
- Voice/video calls
- ...

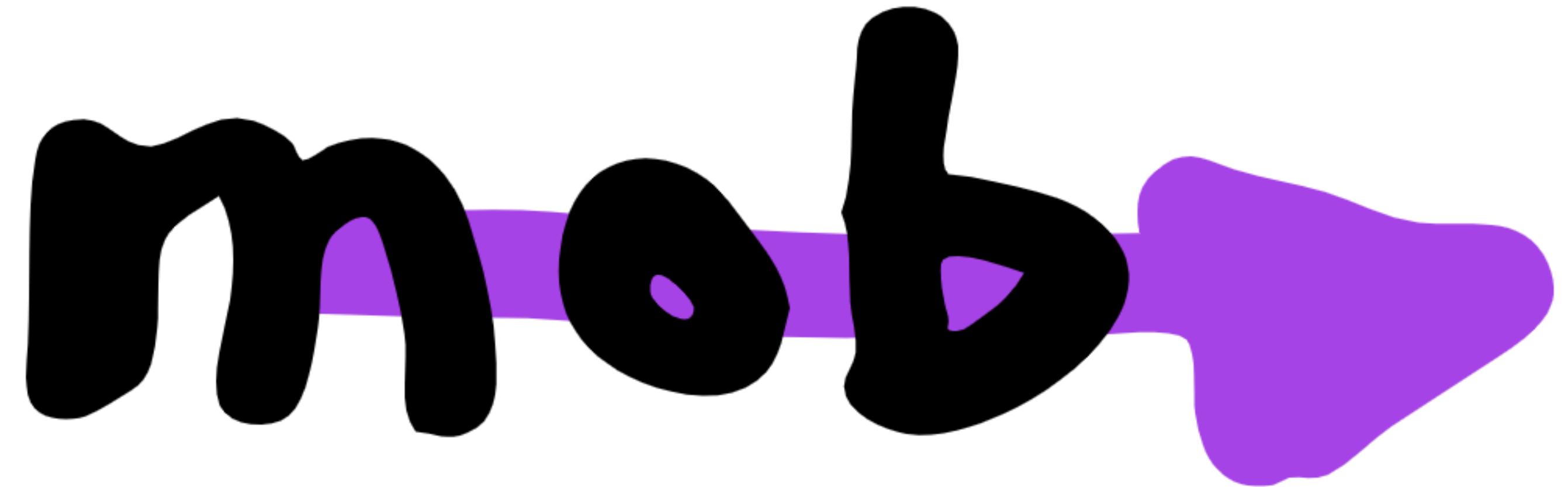
# Pair/Mob Programming Tool

<https://mob.sh>

Entwickelt von  
Dr. Simon Harrer

- mob start
- mob next
- mob done

Fast git handover with mob



[build passing](#) [contributors 36](#) [downloads 6.3k](#) [downloads@latest 151](#) [release v1.3.0](#) [stars 400](#)

Smooth [git handover](#) with 'mob'.

- [mob is an open source command line tool written in go](#)
- [mob is the fastest way to hand over code via git and feels ubersmooth](#)
- [mob supports remote mob or pair programming using screen sharing](#)
- [mob works on every platform, even Apple Silicon](#)
- [mob keeps your branches clean and only creates WIP commits on temporary wip branches](#)
- [mob supports multiple wip branches per base branch](#)
- [mob notifies you when it's time ⏱ to handover](#)
- [mob can moo 🐄](#)

# Beispiel Session

```
→ ~/Workspace/Private/aoc-2020 git:(master) mob start 10
  git fetch origin --prune
  git merge FETCH_HEAD --ff-only
> starting new mob session from origin/master
  git checkout -B mob-session origin/master
  git push --no-verify --set-upstream origin mob-session
> 10 minutes timer started (finishes at approx. 23:19)
> you are mob programming
> on wip branch mob-session (base branch master)
→ ~/Workspace/Private/aoc-2020 git:(mob-session)
```

```
→ ~/Workspace/Private/aoc-2020 git:(mob-session) mob done
  git fetch origin --prune
  git push --no-verify origin mob-session
  git checkout master
  git merge origin/master --ff-only
  git merge --squash --ff mob-session
  git branch -D mob-session
  git push --no-verify origin --delete mob-session
README.md | 2 ++
1 file changed, 2 insertions(+)
👉 To finish, use
git commit
```

```
→ ~/Workspace/Private/aoc-2020 git:(mob-session) x mob next
  git add --all
  git commit --message mob next [ci-skip] --no-verify
  git push --no-verify origin mob-session
README.md | 2 ++
1 file changed, 2 insertions(+)
→ ~/Workspace/Private/aoc-2020 git:(mob-session)
```

```
→ ~/Workspace/Private/aoc-2020 git:(master) x gst
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified: README.md

→ ~/Workspace/Private/aoc-2020 git:(master)
```

# Zeichnen, Skizzieren, Boards

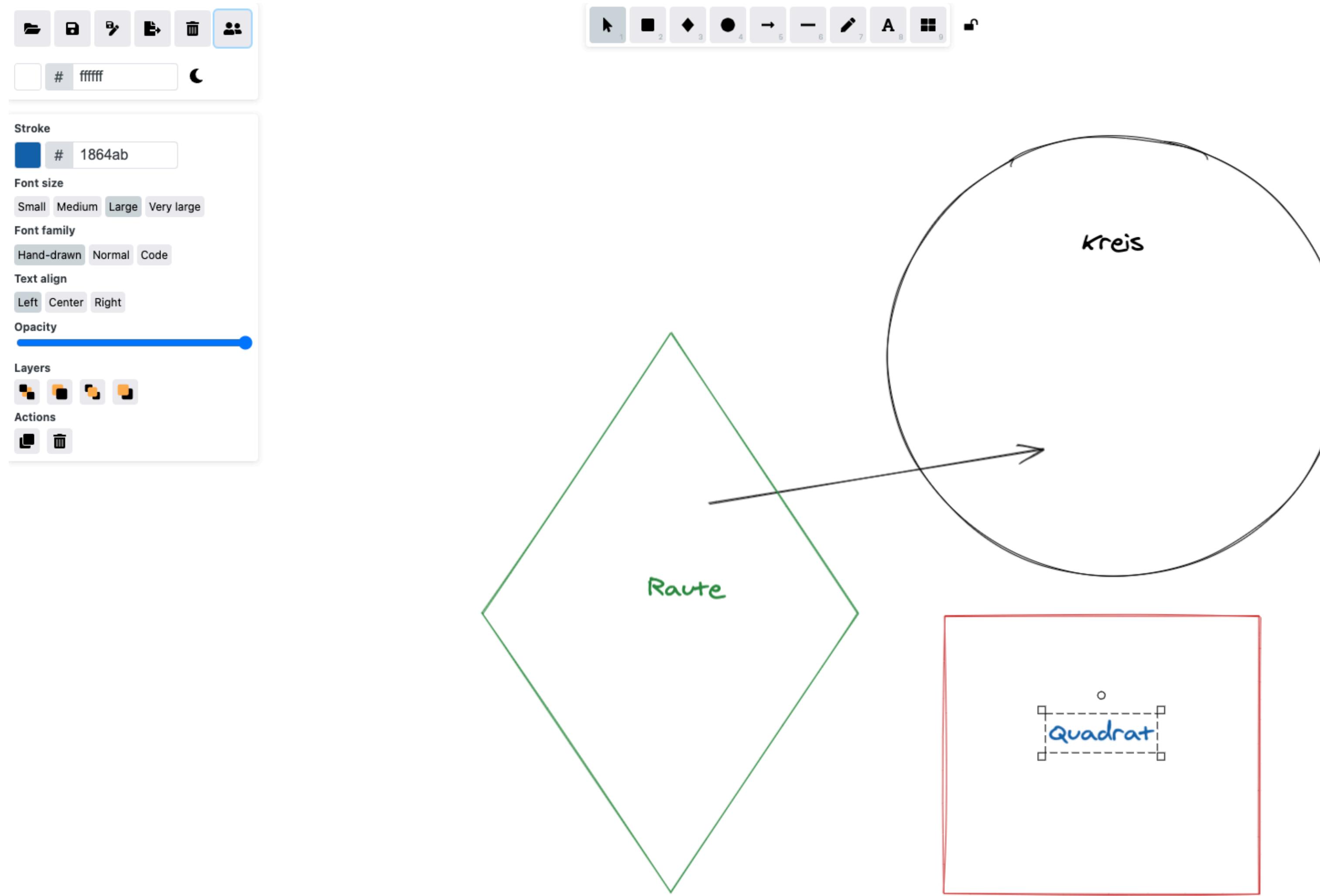
Miro Whiteboard (<https://miro.com/>)

- Registrierung
- App oder Browser
- viele Templates für Charts, Agile Boards, Sticky Notes, Mind Maps, etc. mit Erklärungen zu den Template
- comments, chats, video chat, screen sharing

Excalidraw (<https://excalidraw.com/>)

- Gut zum Zeichnen und Skizzieren
- Keinerlei Registrierung notwendig
- Einladung anderer per Link möglich
- Export von Zeichnungen

# Beispiel Excalidraw



# Tools für Agile

Confluence

Jira

Trello

EasyRetro (<https://easyretro.io/>)

Retromat (<https://retromat.org/>) by  
Corinna Baldauf

# More Tools

Big Blue Button (seht ihr gerade in Aktion)

Hopin - Event Platform, Konferenz

Gather.town - Konferenz

# Den Tag strukturieren

Arbeitsbeginn

Pausen

Feierabend

# Etikette für Remote Zusammenarbeit

## “Einchecken”

Abwesenheit/“nicht gestört werden wollen” kenntlich machen

keine sofortige Reaktion auf schriftliche Kommunikation erwarten

bei Calls Webcam einschalten

ausreden lassen, Latenzzeiten mit einkalkulieren

lieber einmal mehr reden/schreiben/nachfragen als zu ruhig sein

Aktionen im Channel/Room ankündigen, Einwände abfragen

schriftliche Kommunikation besser über Channels oder Rooms und in Threads,  
weniger über direkte Nachrichten

beachten, dass die non-verbale Kommunikation sehr eingeschränkt ist

auch mal nach dem Befinden fragen

Nach Corona ist vor ???

# Vielen Dank

 [mail@birgitkratz.de](mailto:mail@birgitkratz.de)

 @bikratz

Icons made by [Freepik](#) from [www.flaticon.com](#)

# Fragen?