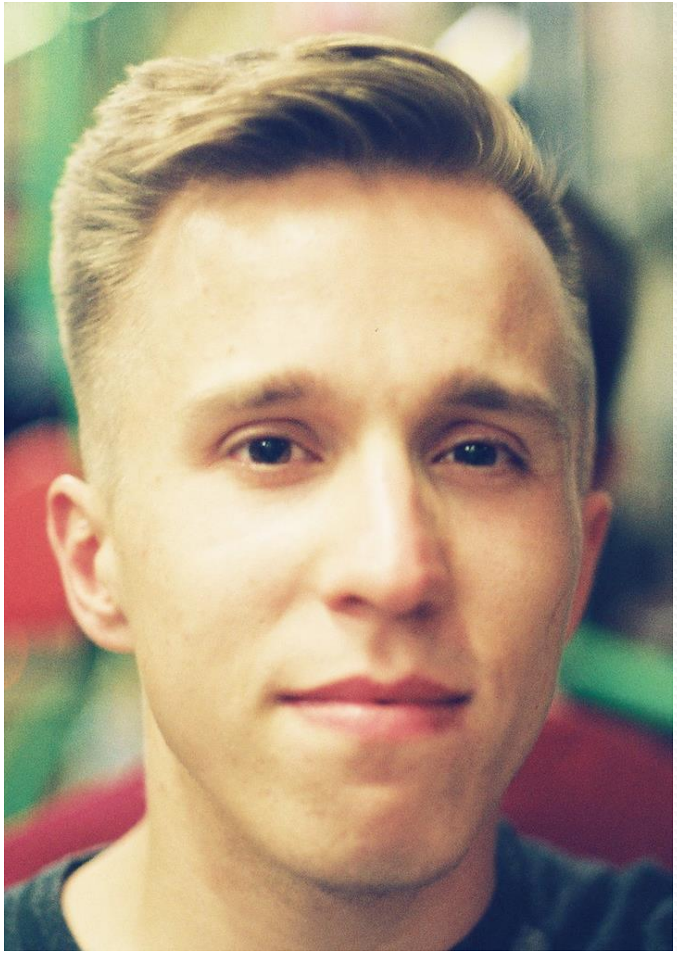
GFT

Cypress

Enabling Faster Execution



$(whoami)

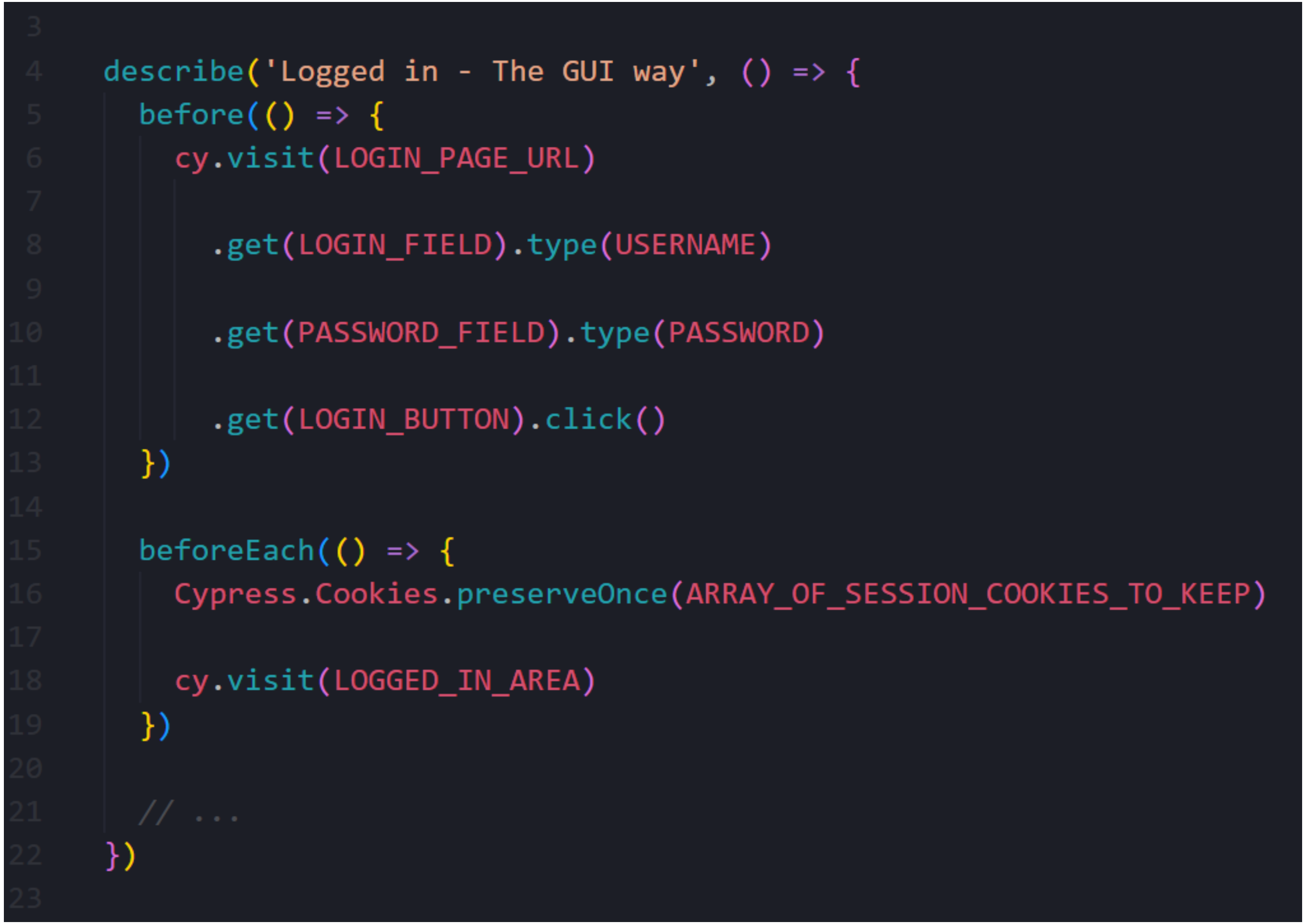
Staszek Rosada

- Senior QA Engineer with 5 years of experience
- Lead QA @ GFT Poland
- Working with Cypress for around 3 years
- Facilitator of Cypress Community @ GFT Poland
- TestOps enthusiast

stop using gui

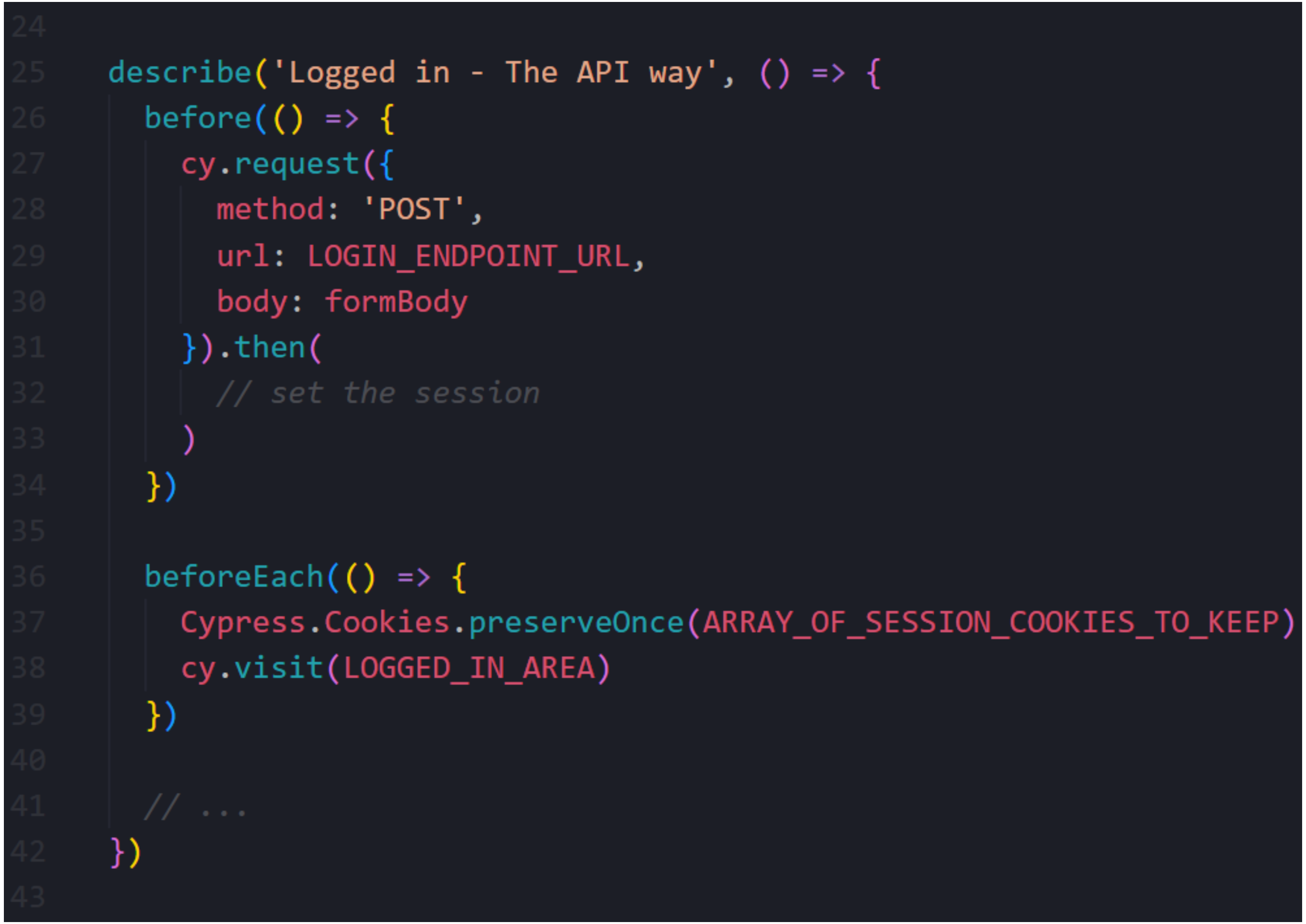
at least so much

gui login

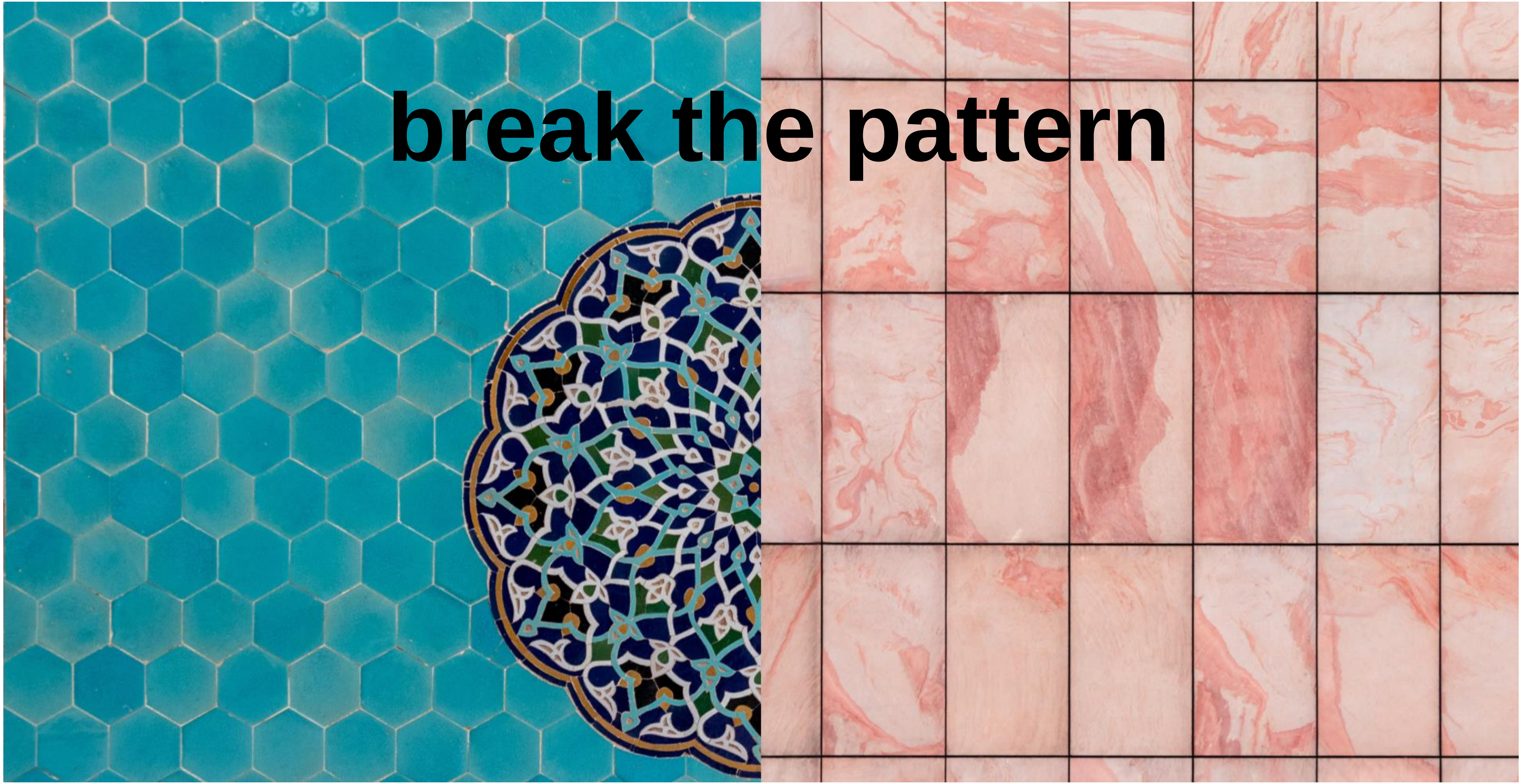


```
describe('Logged in - The GUI way', () => {
  before(() => {
    cy.visit(LOGIN_PAGE_URL)

      .get(LOGIN_FIELD).type(USERNAME)

      .get(PASSWORD_FIELD).type(PASSWORD)

      .get(LOGIN_BUTTON).click()
  })

  beforeEach(() => {
    Cypress.Cookies.preserveOnce(ARRAY_OF_SESSION_COOKIES_TO_KEEP)

    cy.visit(LOGGED_IN_AREA)
  })

  // ...
})
```

api login

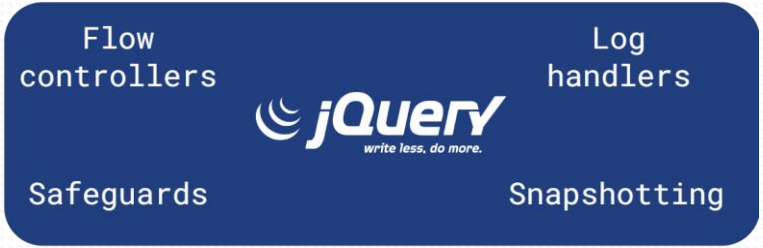


```
describe('Logged in - The API way', () => {
  before(() => {
    cy.request({
      method: 'POST',
      url: LOGIN_ENDPOINT_URL,
      body: formBody
    }).then(
      // set the session
    )
  })

  beforeEach(() => {
    Cypress.Cookies.preserveOnce(ARRAY_OF_SESSION_COOKIES_TO_KEEP)
    cy.visit(LOGGED_IN_AREA)
  })

  // ...
})
```



break the pattern

How do Cypress commands work?



Flow controllers

Log handlers

Safeguards

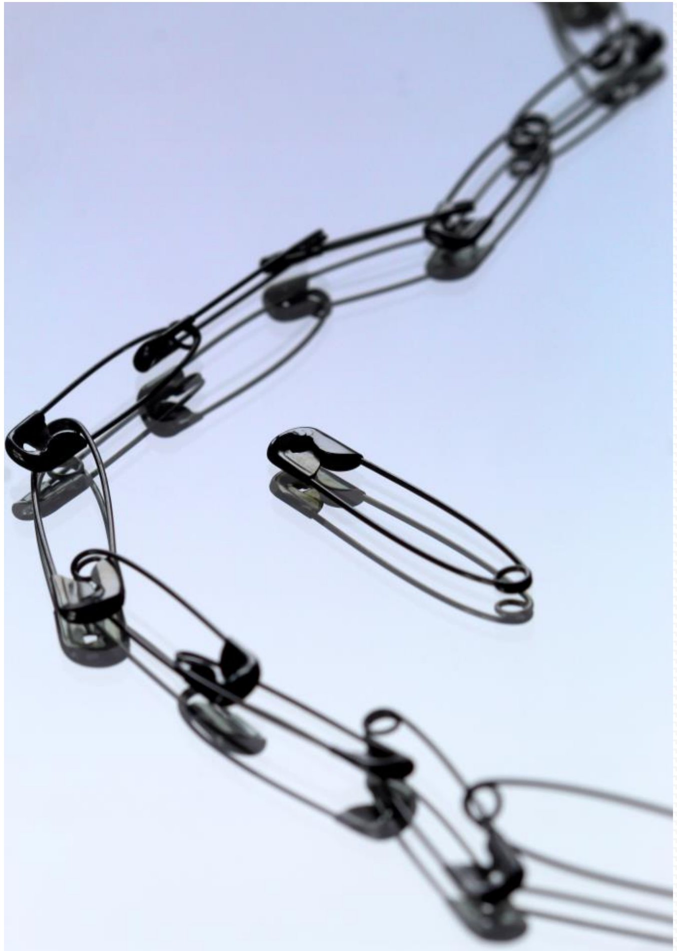
Snapshotting

upsides

- auto wait
- DOM history
- retryability
- readability

downsides

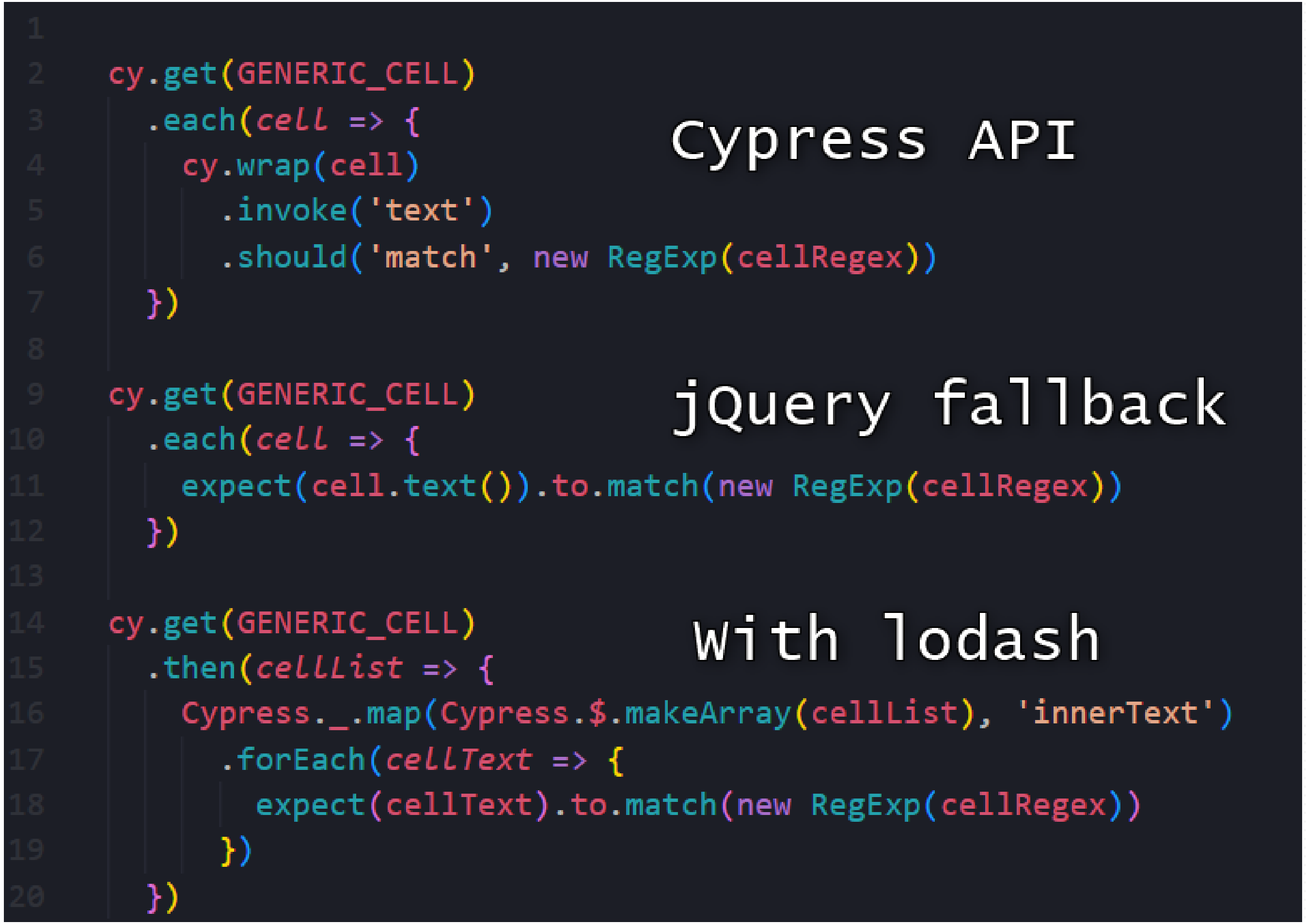
- time
- memory consumption



Be less dependent on Cypress

- Understand how your application behaves
- Identify the safeguards yourself
- Use jQuery and plain JS instead

different ways



```
cy.get(GENERIC_CELL)
  .each(cell => {
    cy.wrap(cell)
      .invoke('text')
      .should('match', new RegExp(cellRegex))
  })

cy.get(GENERIC_CELL)
  .each(cell => {
    expect(cell.text()).to.match(new RegExp(cellRegex))
  })

cy.get(GENERIC_CELL)
  .then(cellList => {
    Cypress._.map(Cypress.$.makeArray(cellList), 'innerText')
      .forEach(cellText => {
        expect(cellText).to.match(new RegExp(cellRegex))
      })
  })
```

Cypress API

jQuery fallback

With lodash

real life peformance comparison

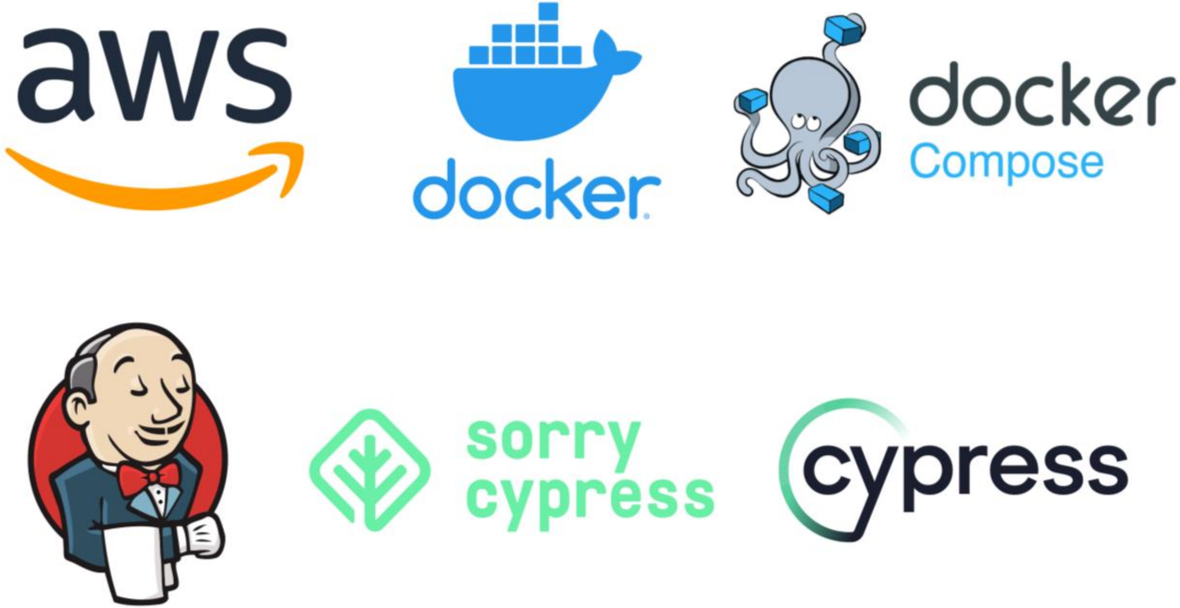
Cypress API	28s
jQuery fallback	4s
With lodash	3s

Overall time gain in project	20 min*

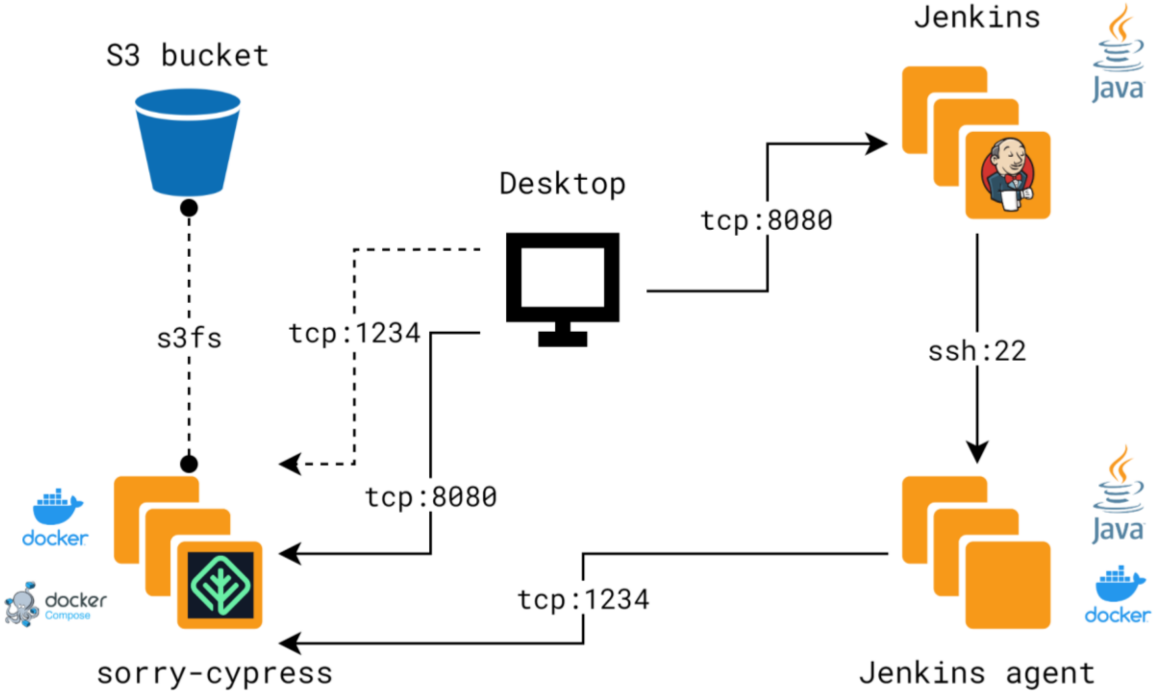
* sum of gains for all changed implementations

distribute work

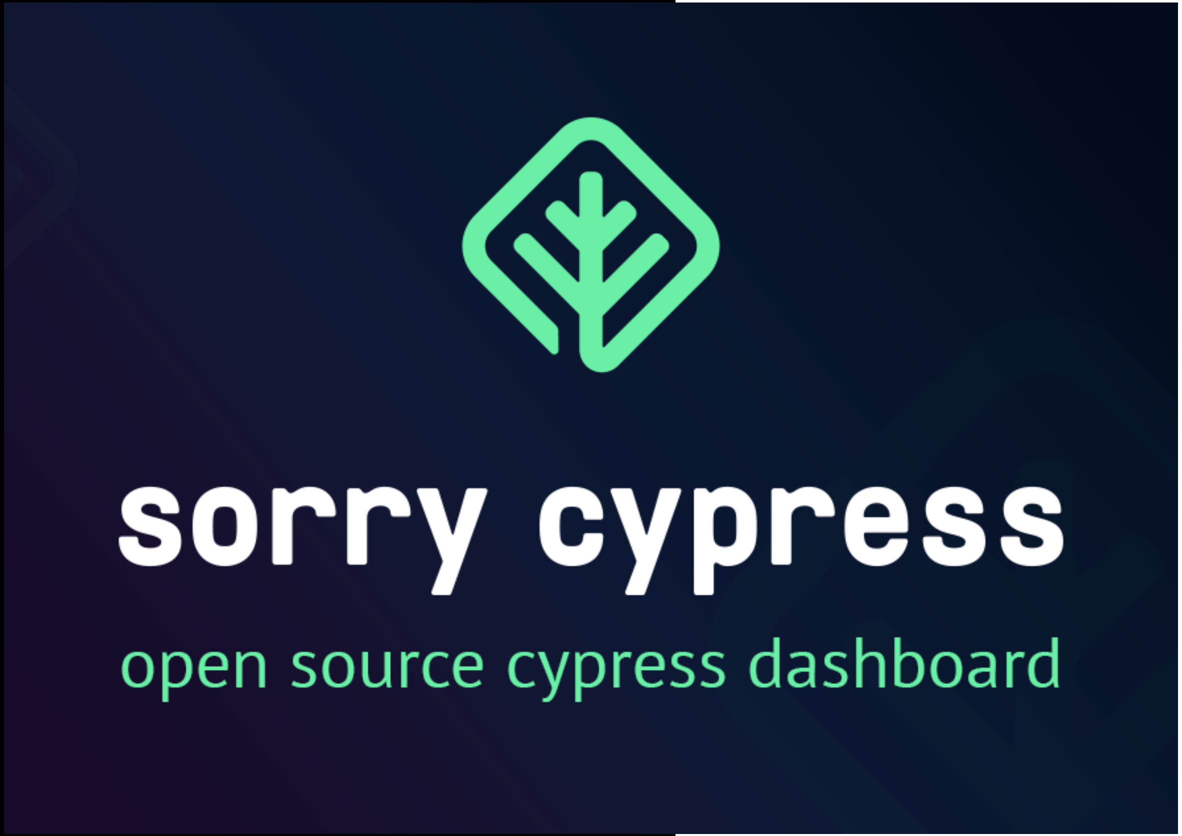
technology overview



architecture overview



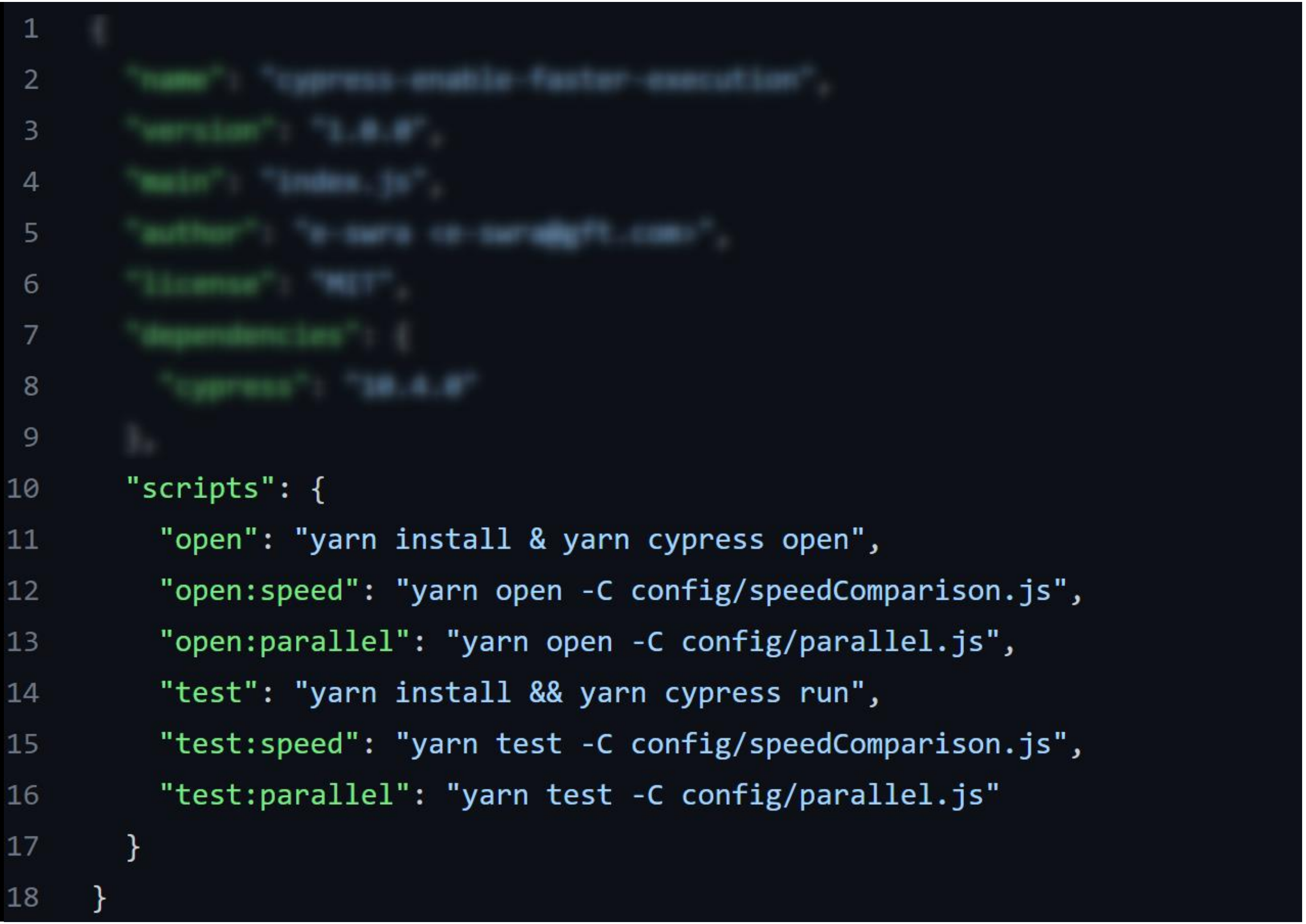
dashboard alternative



code prep

npm/yarn scripts

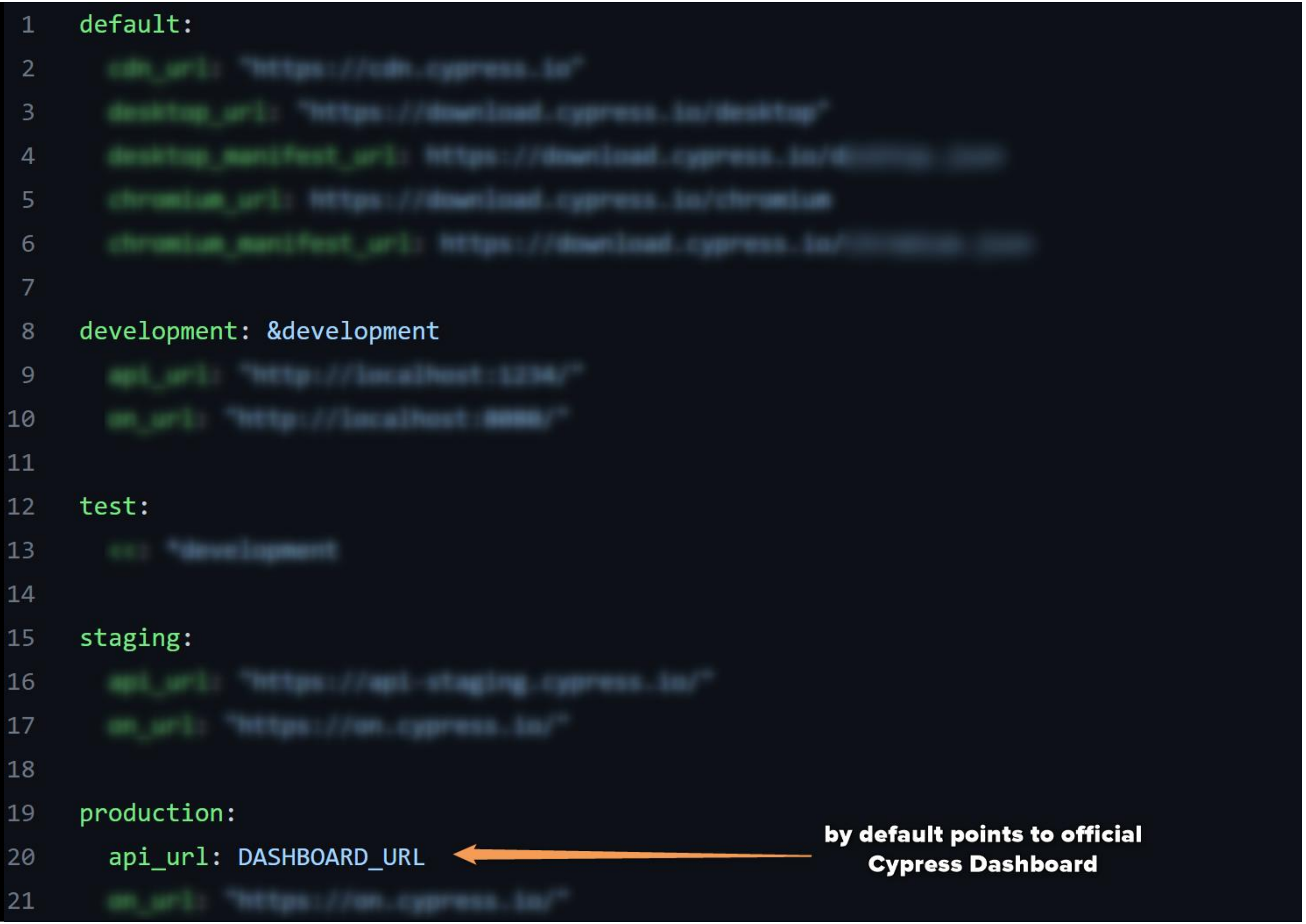
concise
docker
run



```
{
[redacted]
  "scripts": {
    "open": "yarn install & yarn cypress open",
    "open:speed": "yarn open -C config/speedComparison.js",
    "open:parallel": "yarn open -C config/parallel.js",
    "test": "yarn install && yarn cypress run",
    "test:speed": "yarn test -C config/speedComparison.js",
    "test:parallel": "yarn test -C config/parallel.js"
  }
}
```

custom image definition

app.yml

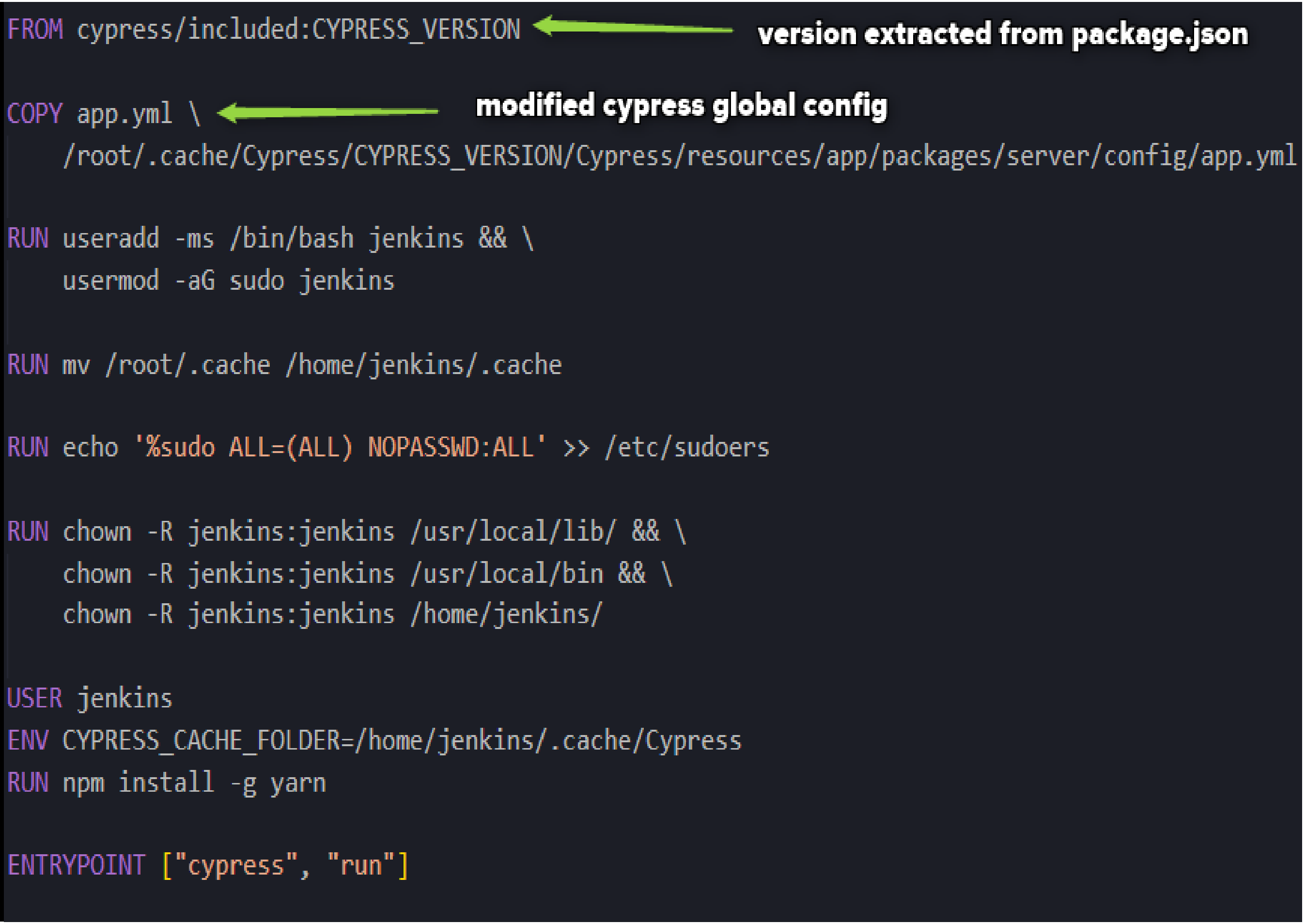


```
default:
  [redacted]

development: &development
  [redacted]

test:
  [redacted]

staging:
  [redacted]

production:
  api_url: DASHBOARD_URL
  [redacted]
```

by default points to official Cypress Dashboard

custom image definition

Dockerfile



```
FROM cypress/included:CYPRESS_VERSION

COPY app.yml \
    /root/.cache/Cypress/CYPRESS_VERSION/Cypress/resources/app/packages/server/config/app.yml

RUN useradd -ms /bin/bash jenkins && \
    usermod -aG sudo jenkins

RUN mv /root/.cache /home/jenkins/.cache

RUN echo '%sudo ALL=(ALL) NOPASSWD:ALL' >> /etc/sudoers

RUN chown -R jenkins:jenkins /usr/local/lib/ && \
    chown -R jenkins:jenkins /usr/local/bin && \
    chown -R jenkins:jenkins /home/jenkins/

USER jenkins
ENV CYPRESS_CACHE_FOLDER=/home/jenkins/.cache/Cypress
RUN npm install -g yarn

ENTRYPOINT ["cypress", "run"]
```

version extracted from package.json

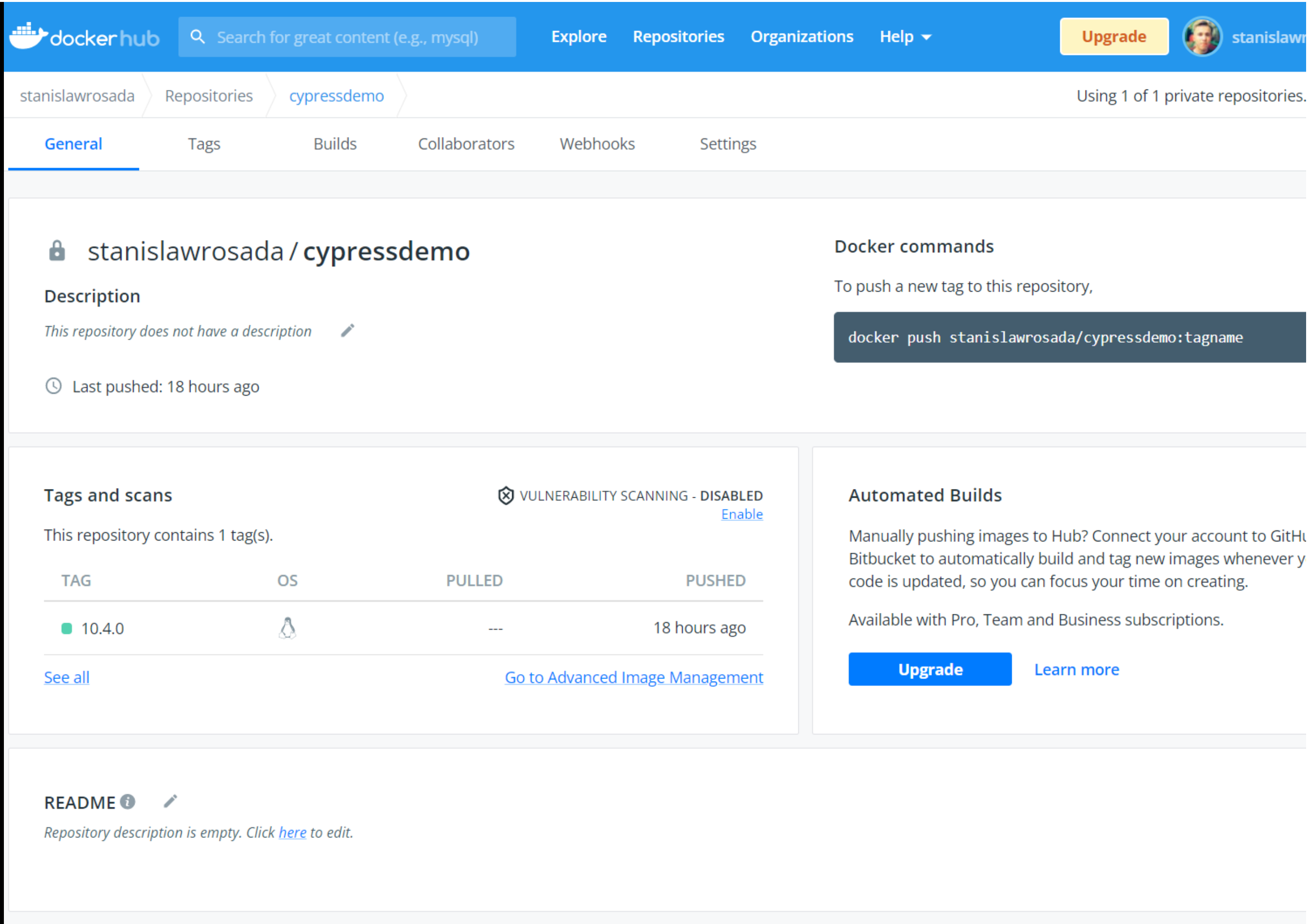
modified cypress global config

CI setup

12 September 2022

private docker repo

choose
your own
registry



Using 1 of 1 private repositories.

stanislawrosada / **cypressdemo**

Description

This repository does not have a description

Last pushed: 18 hours ago

Docker commands

To push a new tag to this repository,

```
docker push stanislawrosada/cypressdemo:tagname
```

Tags and scans

VULNERABILITY SCANNING - **DISABLED**
Enable

This repository contains 1 tag(s).

TAG	OS	PULLED	PUSHED
🟩 10.4.0	🐧	---	18 hours ago

See all

Go to Advanced Image Management

Automated Builds

Manually pushing images to Hub? Connect your account to GitHu
Bitbucket to automatically build and tag new images whenever y
code is updated, so you can focus your time on creating.

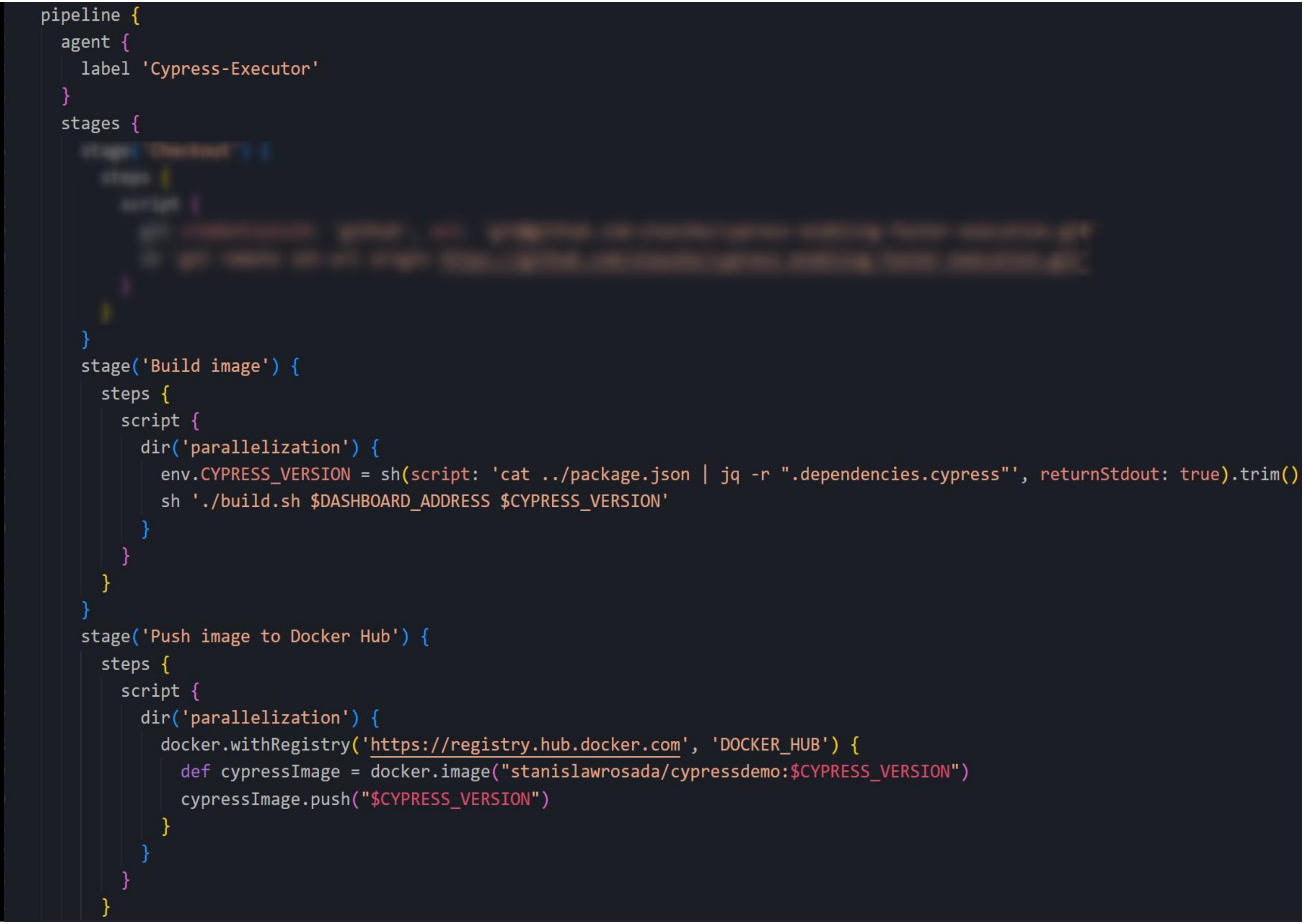
Available with Pro, Team and Business subscriptions.

README

Repository description is empty. Click here to edit.

private docker repo

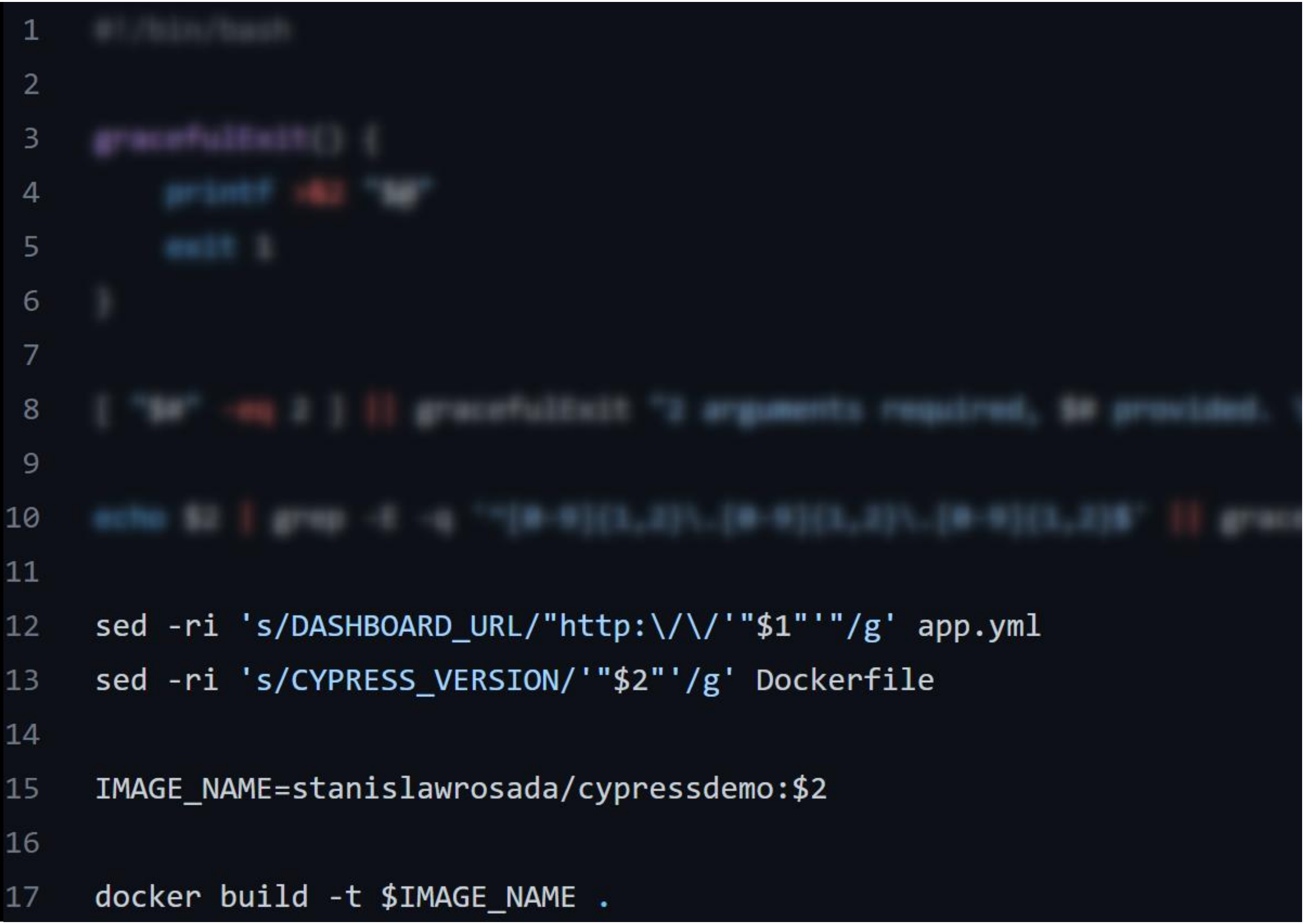
custom
image
build &
push



```
pipeline {
  agent {
    label 'Cypress-Executor'
  }
  stages {
    [redacted]
    }
    stage('Build image') {
      steps {
        script {
          dir('parallelization') {
            env.CYPRESS_VERSION = sh(script: 'cat ../package.json | jq -r ".dependencies.cypress"', returnStdout: true).trim()
            sh './build.sh $DASHBOARD_ADDRESS $CYPRESS_VERSION'
          }
        }
      }
    }
    stage('Push image to Docker Hub') {
      steps {
        script {
          dir('parallelization') {
            docker.withRegistry('https://registry.hub.docker.com', 'DOCKER_HUB') {
              def cypressImage = docker.image("stanislawrosada/cypressdemo:$CYPRESS_VERSION")
              cypressImage.push("$CYPRESS_VERSION")
            }
          }
        }
      }
```

custom image build

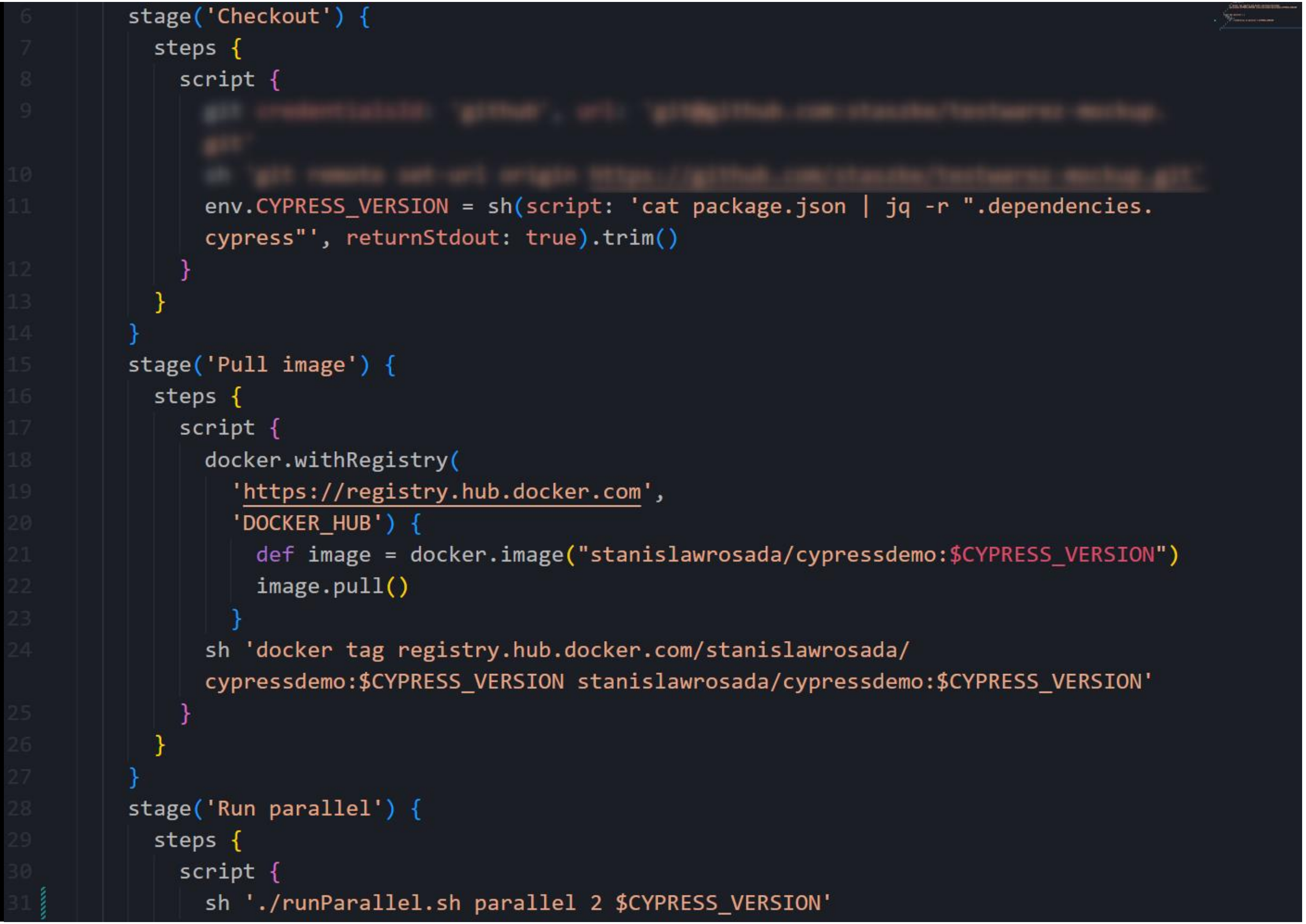
build
script



```
#!/bin/bash

gracefulExit() {
    printf >&2 "$@"
    exit 1
}

[ "$#" -eq 2 ] || gracefulExit "2 arguments required, $# provided. \

echo $2 | grep -E -q '^[0-9]{1,2}\.[0-9]{1,2}\.[0-9]{1,2}$' || grace

sed -ri 's/DASHBOARD_URL/"http:\/\/'"$1"'"/g' app.yml
sed -ri 's/CYPRESS_VERSION/'"$2"'/g' Dockerfile

IMAGE_NAME=stanislawrosada/cypressdemo:$2

docker build -t $IMAGE_NAME .
```

running in jenkins

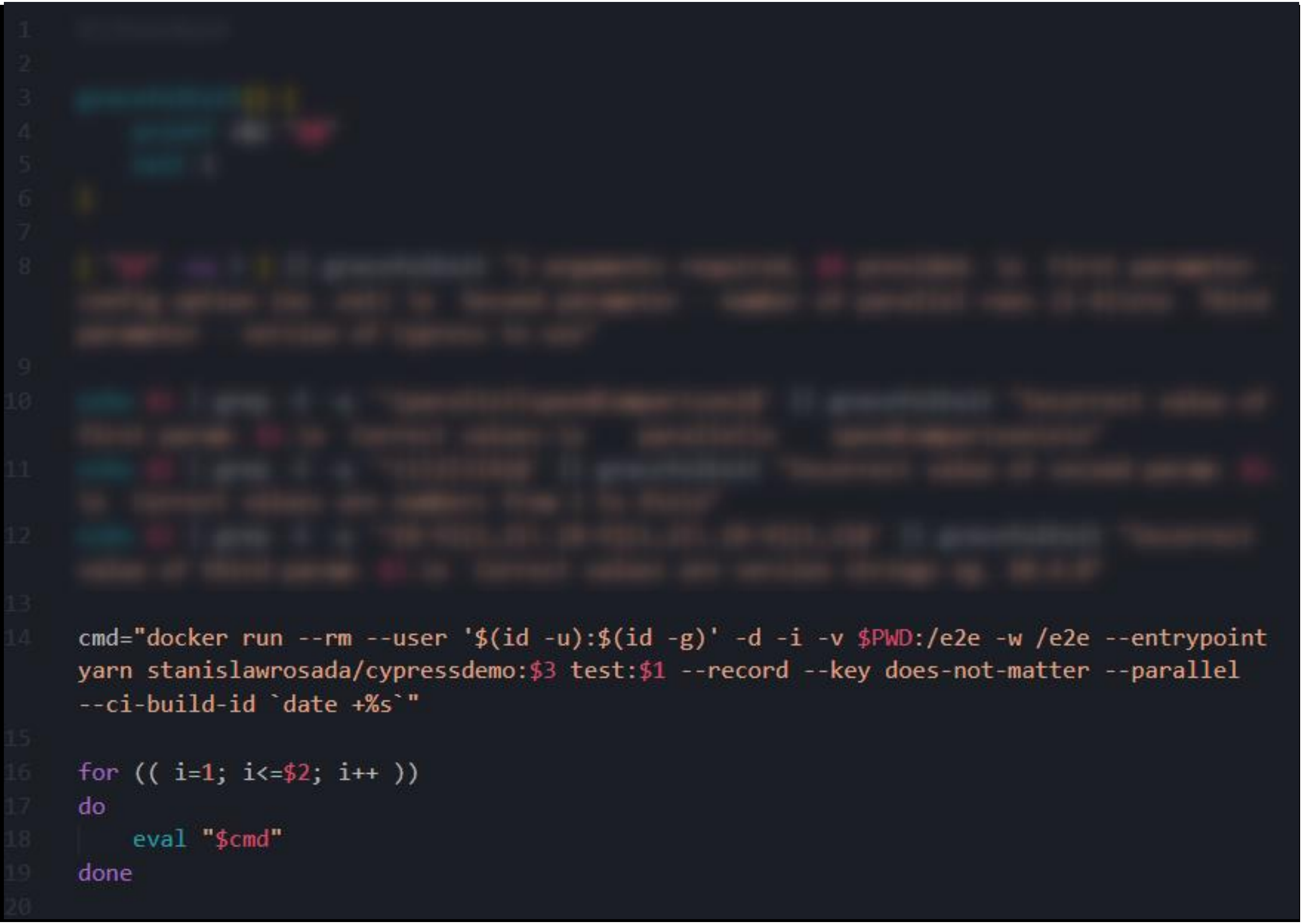
execution pipeline



```
    stage('Checkout') {
      steps {
        script {
          [redacted]
          env.CYPRESS_VERSION = sh(script: 'cat package.json | jq -r ".dependencies.cypress"', returnStdout: true).trim()
        }
      }
    }
    stage('Pull image') {
      steps {
        script {
          docker.withRegistry(
            'https://registry.hub.docker.com',
            'DOCKER_HUB') {
              def image = docker.image("stanislawrosada/cypressdemo:$CYPRESS_VERSION")
              image.pull()
            }
          sh 'docker tag registry.hub.docker.com/stanislawrosada/cypressdemo:$CYPRESS_VERSION stanislawrosada/cypressdemo:$CYPRESS_VERSION'
        }
      }
    }
    stage('Run parallel') {
      steps {
        script {
          sh './runParallel.sh parallel 2 $CYPRESS_VERSION'
```

running in jenkins

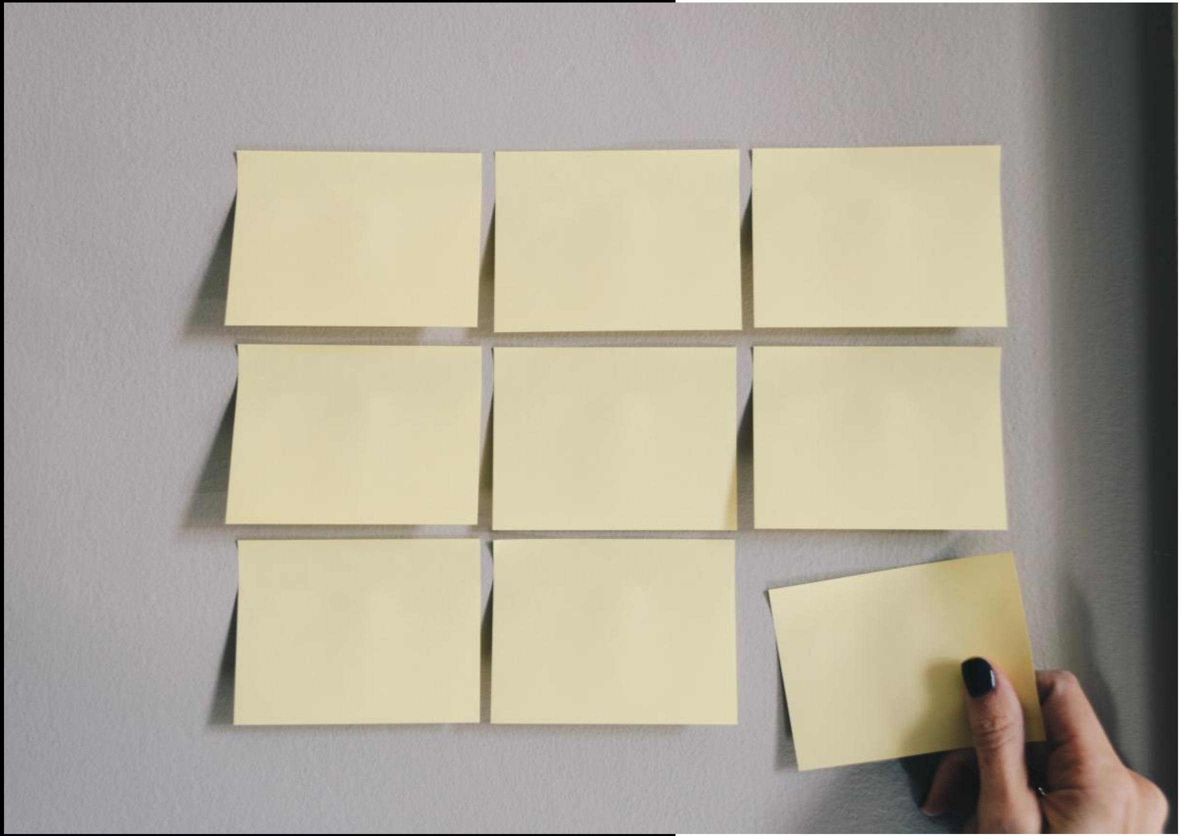
execution script



[redacted]

```
cmd="docker run --rm --user '$(id -u):$(id -g)' -d -i -v $PWD:/e2e -w /e2e --entrypoint yarn stanislawrosada/cypressdemo:$3 test:$1 --record --key does-not-matter --parallel --ci-build-id `date +%s`"

for (( i=1; i<=$2; i++ ))
do
    eval "$cmd"
done
```

conclusion



thank you

/staszke

/stan-ros

stanislaw.rosada@gmail.com