



SNAPSHOTS THROUGH TIME

How we evolved our screenshot tests

THE PLAN

01

WHY SCREENSHOTS?

The problem they solved

02

FIRST STEPS

Initial implementation

03

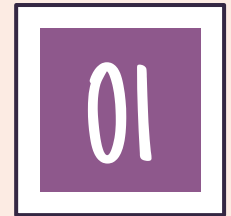
SPEED & RELIABILITY

Reducing flakes and addressing slowness

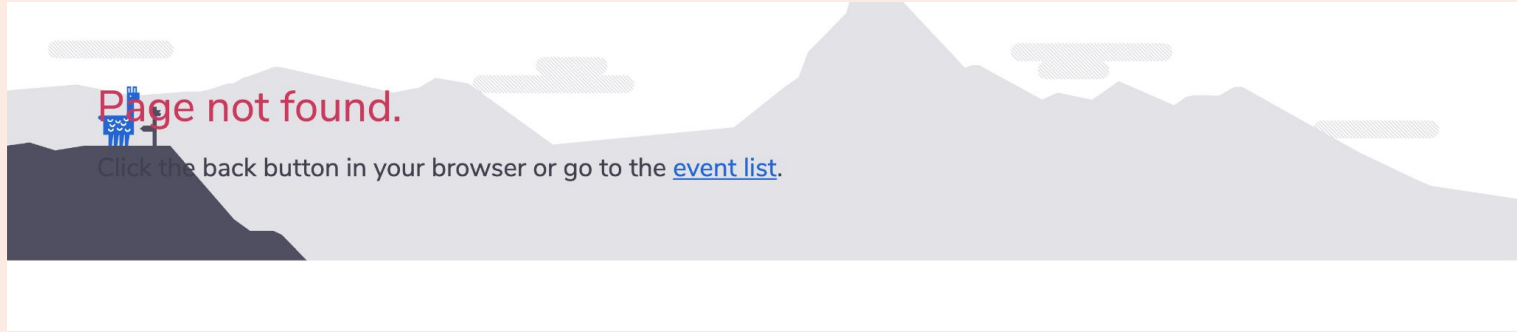
04

LOOKING AHEAD


Where might we go from here?



WHY SCREENSHOTS?







New Password 

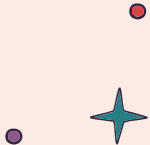
Use at least 8 characters

Reset Password

You can use letters, numbers, and symbols! 

Untitled Collection 

0



FRONTEND TESTING LAYERS



UNIT TESTS

Jest, Enzyme and React
Testing Library



LINTING

ESLint and Stylelint



END-TO-END TESTS

Cypress

UNIT TESTS

```
it( name: 'should disable next and prev buttons when total number of pages is one', fn: () :void => {  
    const pagination :PaginationTestHelper = PaginationTestHelper.render( props: { totalNumberOfPages: 1 })  
  
    expect(pagination.nextDisabled()).toBeTrue()  
    expect(pagination.prevDisabled()).toBeTrue()  
})
```

UNIT TESTS

```
it( name: 'should add class for trigger hover when triggerHover property specified', fn: () :void => {  
    const dropdown : DropdownTestHelper = DropdownTestHelper.render( props: { triggerHover: true })  
  
    expect(dropdown.classExists( className: 'dropdown__button--hover')).toBe( expected: true)  
})
```

LINTING

```
4  $white: #ffffff;
5  $grey-0: #fbfbfd;
6  $grey-1: #f4f4f6;
7  $grey-2: #e3e3e7;
8  $grey-3: #ccccd4;
9  $grey-4: #b7b7c12;
10  $grey-5: #a0
11  $grey-6: #8a
12  $grey-7: #757589;
13  $grey-8: #5e5e76;
14  $grey-9: #4f4f63;
```

Stylelint: Unexpected invalid hex color "#b7b7c12" (color-no-invalid-hex) ⋮
Stylelint: Fix current file Alt+Shift+Enter More actions... Alt+Enter

```
18
19 .toast__body {
20   white-space: nowrap;
21   overflow: hidden;
22   text-overflow: ellipsis;
23   text-align: center;
24   align: center;
25 }
26 .Toastify__toast--warning,
27 .Toastify__toast--error {
28
```

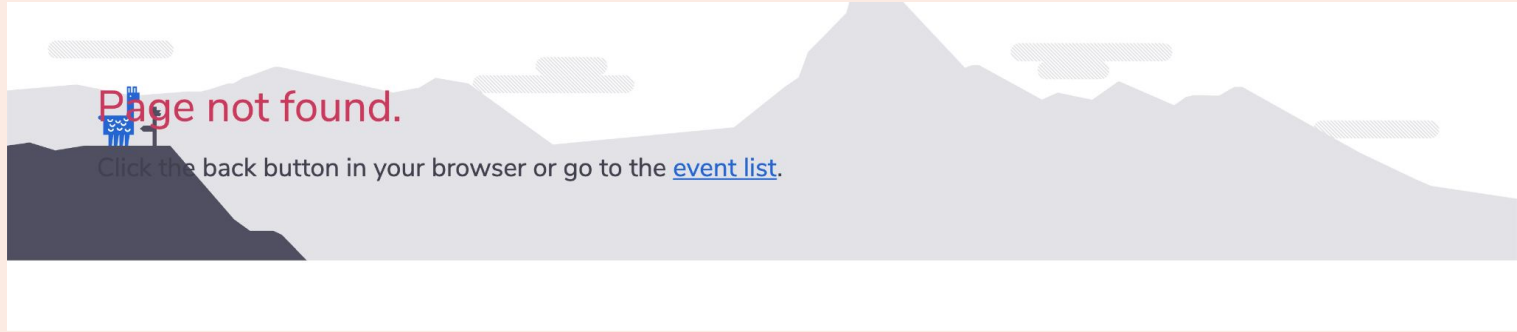
Stylelint: Unexpected unknown property "align" (property-no-unknown) ⋮
Stylelint: Fix current file Alt+Shift+Enter More actions... Alt+Enter

END-TO-END TESTS

```
describe( title: 'Page not Found', fn: () :void => {
    CustomSetup.withDefaultUser(
        name: 'should show 404 page on unrecognised URLs, and take you back to the events list.',
        testBlock: ({ email :EmailAddress }) :void => {
            SelectEventsPage.loginAndVisitEventsPage(email)

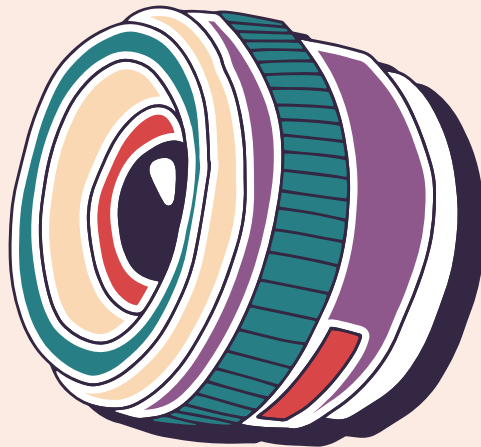
            PageNotFound.getLost()
            PageNotFound.verifyIsShown()
            PageNotFound.goToEventsList()

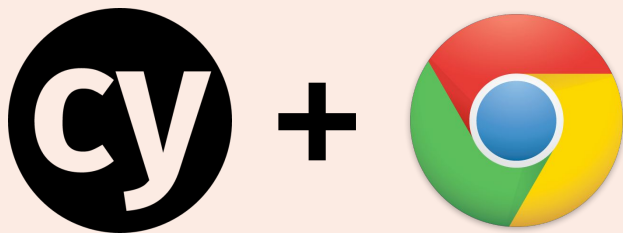
            SelectEventsPage.verifyIsLoaded()
        },
    )
})
```



02

FIRST STEPS





1. Run just like existing end-to-end tests, driven via Cypress.
2. Add a custom Cypress command to take a screenshot and compare to a “canned” image already in source control.
3. Need an update mode to allow us to generate new screenshots and update existing ones.
4. Need to be able to see the diff easily on failure, to diagnose what went wrong.

```
'match-screenshot': async args => {
  const { screenshotPath, name } = args

  const diffsPath :string = `${__dirname}/../diffs`
  const screenshotDir :string = screenshotPath.substring(0, screenshotPath.lastIndexOf('/') + 1)
  const expectedScreenshot :string = `${diffsPath}/${name}.png`

  if (fs.existsSync(expectedScreenshot) && config.env.UPDATE_SNAPSHOTS !== true) {
    const diff = new BlinkDiff({
      imageAPath: expectedScreenshot,
      imageBPath: screenshotPath,
      thresholdType: BlinkDiff.THRESHOLD_PIXEL,
      threshold: 0.01,
      imageOutputPath: `${screenshotDir}/output-${name}.png`,
      delta: 5,
      debug: true,
    })

    const { code } = diff.runSync()
    return diff.hasPassed(code)
  } else {
    // New or Updated Snapshot should be set
    console.log('Ignoring old screenshot, setting screenshot as expected')
    fs.renameSync(screenshotPath, expectedScreenshot)
    return true
  }
}
```

← Compare mode


← Update mode

```
it('view organisation page and compare screenshot', function () :void {
  const filterSelectors :string[] = ['.pagination']
  cy.login(this.screenshotUser.email, this.screenshotUser.password)



  cy.intercept(`${Cypress.env('ADMIN_URL')}/api/organisations`, { fixture: 'organisations.json' })
  cy.visitAdmin( path: '/organisations')


  search( query: 'this-email-should-never-be-used@blah.com')
  blur()
  cy.matchScreenshot( name: 'admin-organisations-page-non-found', options: { ...viewport, blackout: filterSelectors })
})
})
```



Organisations Users 

[Create Organisation](#)

 Show Expired 

Actions	Name 	Account Manager	Expires	Admins	User Limit	Invited Users	Registered Users	Remaining Users
---------	--	-----------------	---------	--------	------------	---------------	------------------	-----------------

No organisations were found matching your search



Post Definition ✕

Mercury Q

Choose a definition to post:

- Mercury (planet)
- Mercury (element)
- Mercury (mythology)
- Mercury (toy manufacturer)
- Mercury Communications
- Mercury Drug
- Mercury Energy
- Mercury Filmworks
- Mercury General
- Mercury Interactive

Post Definition ✕

Mercury Q

Choose a definition to post:

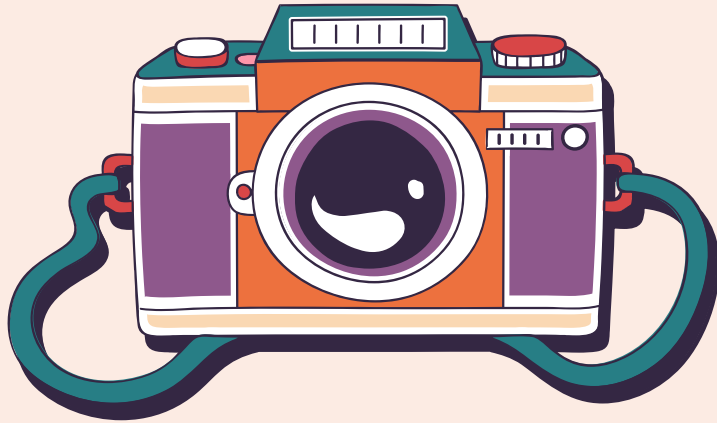
- Mercury (planet)
- Mercury (element)
- Mercury (mythology)
- Mercury (toy manufacturer)
- Mercury Communications
- Mercury Drug
- Mercury Energy
- Mercury Filmworks
- Mercury General
- Mercury Interactive

Post Definition ✕

Mercury Q

Choose a definition to post:

- Mercury (planet)
- Mercury (element)
- Mercury (mythology)
- Mercury (toy manufacturer)
- Mercury Communications
- Mercury Drug
- Mercury Energy
- Mercury Filmworks
- Mercury General
- Mercury Interactive

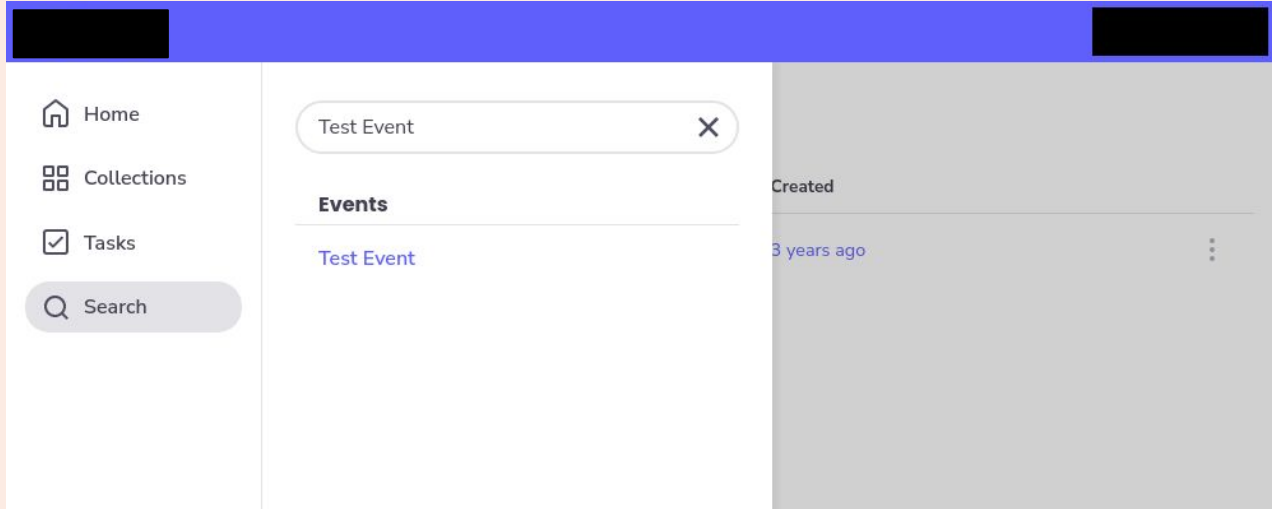


03

SPEED & RELIABILITY

BLACKOUTS

<https://docs.cypress.io/api/commands/screenshot>



FOCUS & HOVER STATE

Change Due Date ×

Jun 19, 2032

June 2032 < >

Mon	Tue	Wed	Thu	Fri	Sat	Sun
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

Change Due Date ×

Jun 19, 2032

June 2032 < >

Mon	Tue	Wed	Thu	Fri	Sat	Sun
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

Change Due Date ×

Jun 19, 2032

June 2032 < >

Mon	Tue	Wed	Thu	Fri	Sat	Sun
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

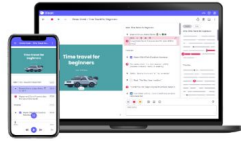
FOCUS & HOVER STATE

```
cy.window().then(async (theWindow) => {  
  ;(theWindow.document?.activeElement as any).blur()  
  cy.root().realHover({ position: 'bottomLeft' })  
})
```

PIXEL TOLERANCE



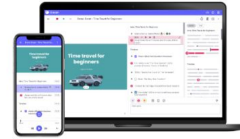
You're all signed up! 🙌



Head to app.glean.co on a supported browser:
[Google Chrome](#) or [Microsoft Edge](#)



You're all signed up! 🙌



Head to app.glean.co on a supported browser:
[Google Chrome](#) or [Microsoft Edge](#)

PIXEL TOLERANCE

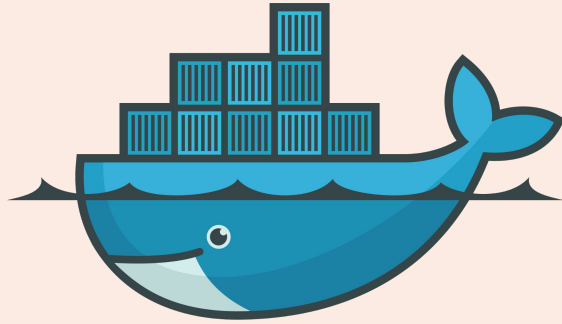
You're all signed up! 🎉



Head to app.glean.co on a supported browser:

[Google Chrome](#) or [Microsoft Edge](#)

LOCAL MACHINE VARIANCE

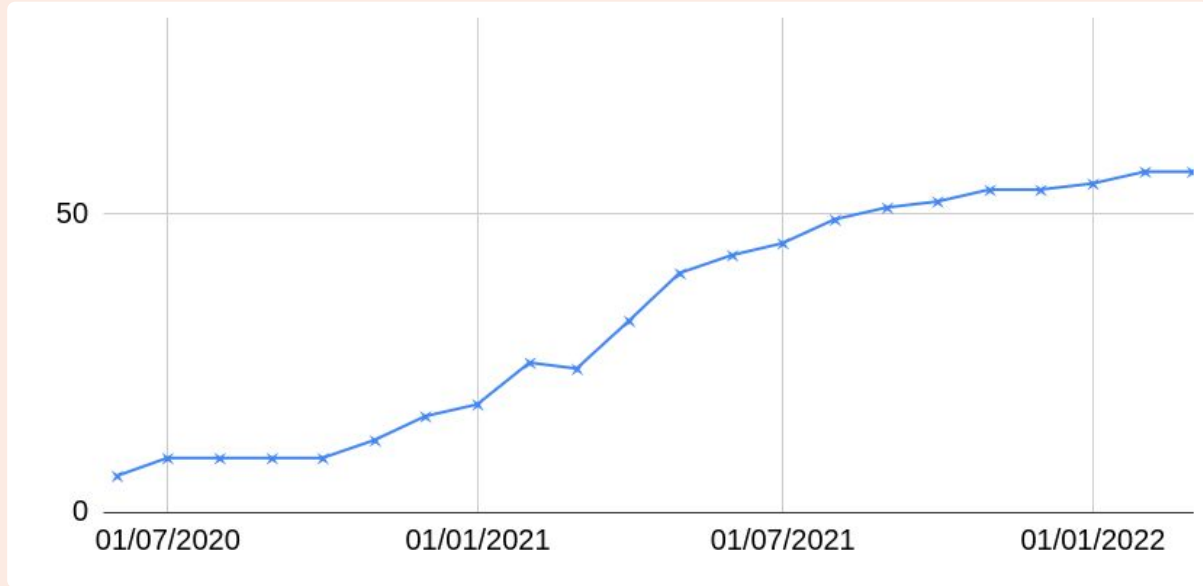


docker

RETRIES

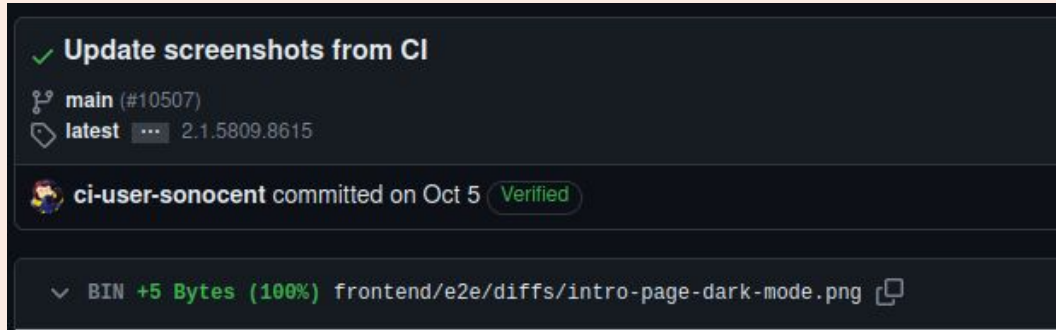
```
.then( fn: () => cy.task( event: 'match-screenshot', arg: { screenshotPath, name })))  
.then( fn: (output: any) => {  
  if (output.success !== true) {  
    if (attempt >= 5) {  
      throw Error('screenshots did not match after 5 retries')  
    }  
  
    Cypress.log( options: {  
      name: `Screenshot ${name}`,  
      message: `Screenshotting again ${output.retryReason}`,  
    })  
  
    // Particularly for when we're generating screenshots, have a small wait between attempts so we can better guarantee  
    // consistency.  
    // eslint-disable-next-line cypress/no-unnecessary-waiting  
    cy.wait(500)  
    return matchScreenshotWithRetries(name, options, attempt: attempt + 1)  
  }  
})
```

SPEED



MITIGATIONS

1. Parallelisation in our CI pipeline.
2. Ability to run a specific test spec locally.
3. Ability to update screenshots from the CI pipeline.



```
✓ Update screenshots from CI  
main (#10507)  
latest ... 2.1.5809.8615  
ci-user-sonocent committed on Oct 5 Verified  
BIN +5 Bytes (100%) frontend/e2e/diffs/intro-page-dark-mode.png
```





Storybook



STORYBOOK - IN BRIEF

```
export default {
  title: 'Screenshot/ForgotPasswordSubmit',
  component: ForgotPassword,
} as Meta<typeof ForgotPassword>

const Template: StoryFn<typeof ForgotPassword> = (args :ForgotPasswordProps & {childre... , { globals :Globals }) =>
  withRouterAndState(globals, <ForgotPassword {...args} />)

export const ForgotPasswordSubmit : AnnotatedStoryFn<ReactRenderer... = Template.bind( thisArg: {} )
ForgotPasswordSubmit.args = {
  loading: false,
  email: 'grover@muppets.ac.uk',
  emailSent: true,
  backButtonUrl: '/',
}
```

STORYBOOK - IN BRIEF

The screenshot displays the Storybook web interface. On the left, a sidebar lists various components under the heading "SCREENSHOT". The main area shows a preview of an "Account" component. The "Account" component has a title bar with a close button. Below the title bar, there are two tabs: "Profile" (selected) and "Plan". The "Profile" tab contains a form with the following fields:

- Preferred Name:** A text input field containing "Cookie Monster".
- Registered Email:** A text input field containing "cookie.monster@muppets.ac.uk".
- Save Changes:** A blue button.

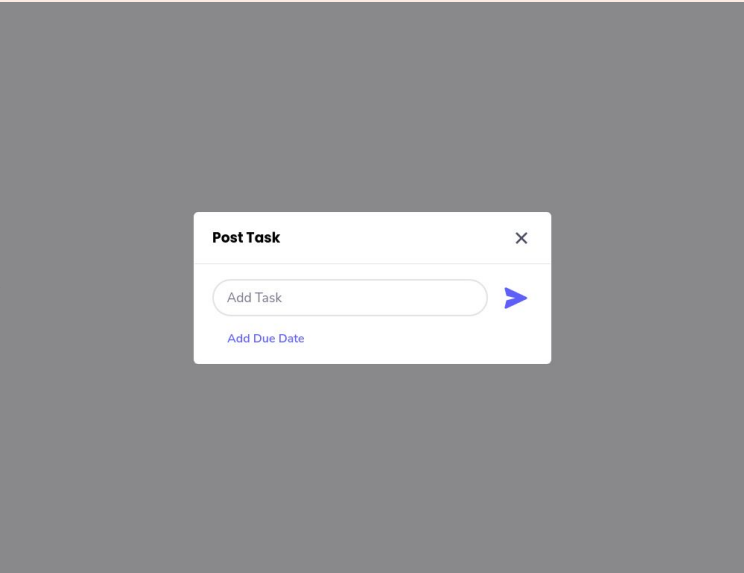
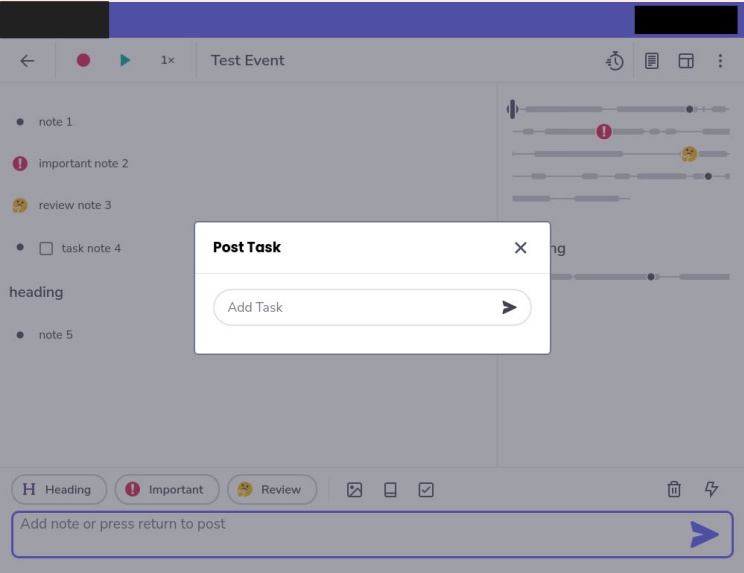
The sidebar on the left lists the following components:

- AudiolImportModal
- AudioOptionsMenu
- BlobDownloadFailureWarning
- BlockedMessage
- Collection
- CollectionsList
- DefinitionModal
- ConfirmDeleteAccountModal
- EarlyAccessModal
- EventsList
- ExpandedView
- FocusTimerAchievementPopup
- FocusTimerModal
- ForgotPassword
- ForgotPasswordSubmit
- ImageSearchModal
- ImportPowerpointModal

OTHER BENEFITS

```
it( title: 'should match screenshot for forgot password', fn: () :void => {  
  cy.visitStory( storyName: 'forgot-password')  
  cy.matchScreenshotAllModes( name: 'forgot-password', options: { height: 720 })  
})
```

ISOLATED COMPONENTS



HOVERING, REVISITED

```
1 import React, { FunctionComponent } from 'react'
2 import styles from './HoverShield.module.scss'
3 import classNames from 'classnames'
4
5 interface HoverShieldProps {
6   enabled: boolean
7   children: React.ReactNode
8 }
9
10 const HoverShield: FunctionComponent<HoverShieldProps> = ({ children, enabled }) => (
11   <div
12     className={classNames(styles.hoverShield, {
13       [styles.hoverShieldEnabled]: enabled,
14     })}
15   >
16     {children}
17   </div>
18 )
19
20 export default HoverShield
```

```
1 .hoverShield {
2   z-index: 100000;
3   height: 100%;
4   width: 100%;
5
6   &Enabled {
7     pointer-events: none;
8   }
9 }
```

LOCAL UPDATES





59

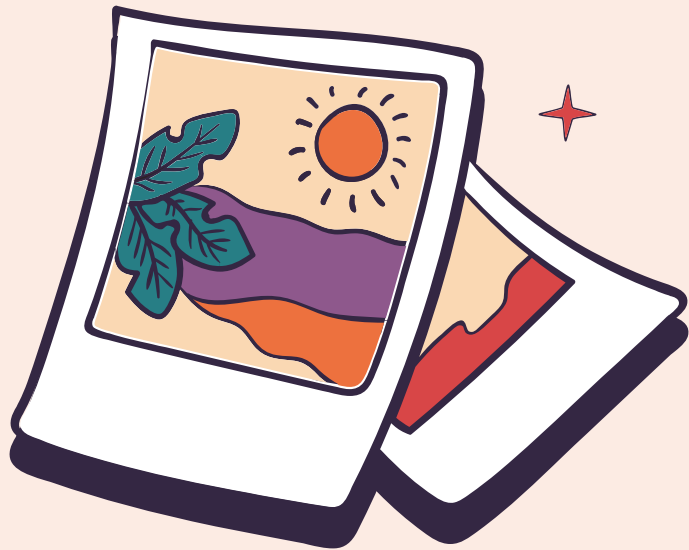
Number of screenshots

12:00

To run in serial in CI

N/A (don't even try)

To run all the tests on a local machine



65

Number of screenshots

2:10

To run in serial in CI

6:00-8:00

To run all the tests on a local machine



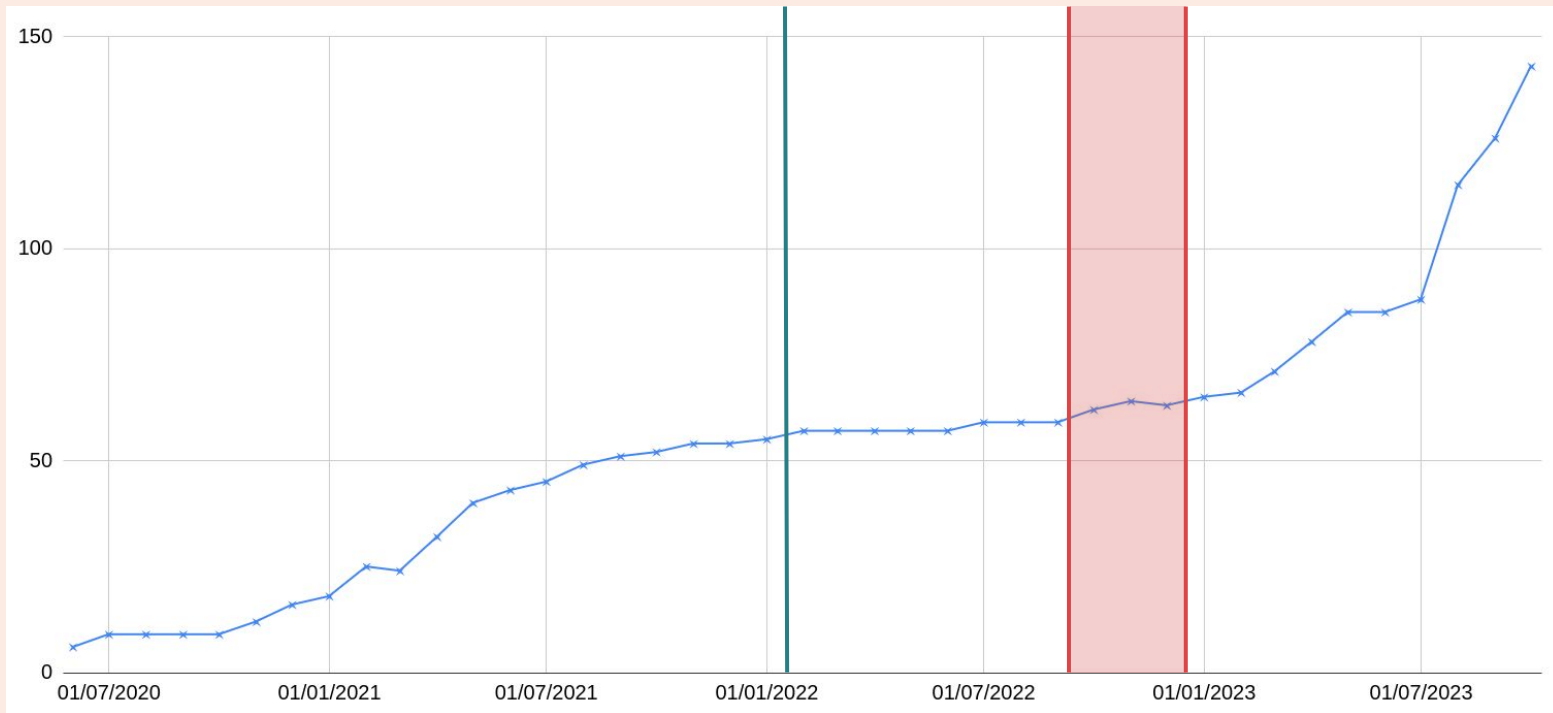
LOOKING AHEAD

Where might we go from here?



Update from CI


Storybook Port







Component Testing

<https://docs.cypress.io/guides/component-testing/overview>

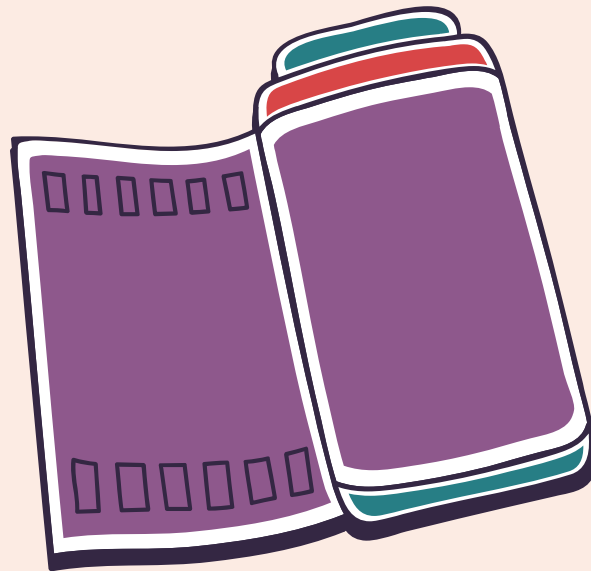


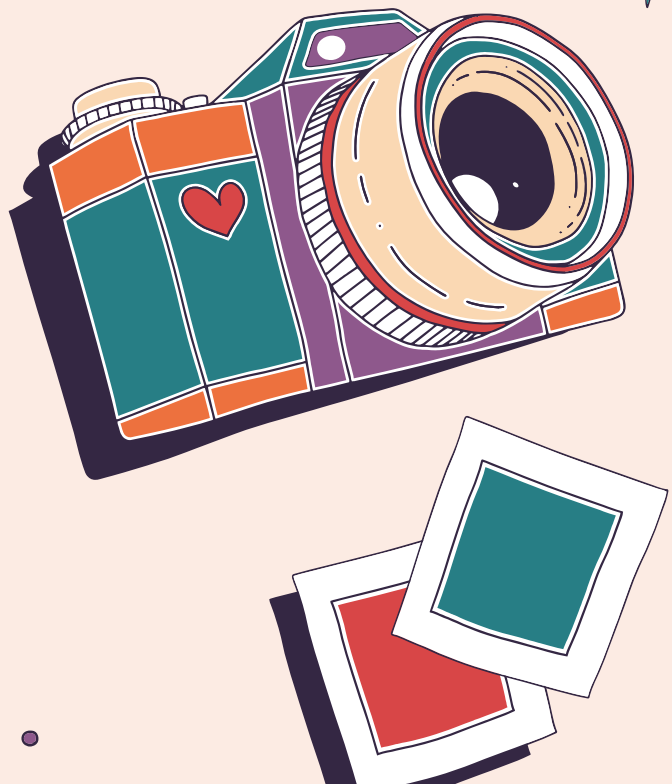
- ```
it('should match screenshot for forgot password', () => {
 cy.mount(<ForgotPasswordSubmit loading={false} emailSent={true} email="grover@muppets.ac.uk" />)
 cy.matchScreenshotAllModes(name: 'forgot-password', options: { height: 720 })
})
```



# SCREENSHOT TESTING - CONCLUSIONS

- Catches **style** regressions that other automated testing can't.
- Run them on the most lightweight thing possible to reduce flakes and improve speed, i.e. not your full app!
- Stamp out any hovering / focus
- Pixel tolerance, consistency checks & retries.





# THANKS!



Do you have any questions?  
[alysa.burlton@gmail.com](mailto:alysa.burlton@gmail.com)

Credits: This presentation template was created by [Slidesgo](#), including icons by [Flaticon](#), and infographics & images by [Freepik](#)

Please keep this slide for attribution