Getting your automated tests in the pipeline

Hugh McCamphill | Test Lead | Glofox | @juegotester

HUSTEF

HUNGARIAN SOFTWARE TESTING FORUM

Tests not in the pipeline don’t make a sound when they fail

12

Software Delivery Performance Metrics

Change lead time

Deployment frequency

Mean Time to Restore

Change failure rate

Deploying continuously reduces risk

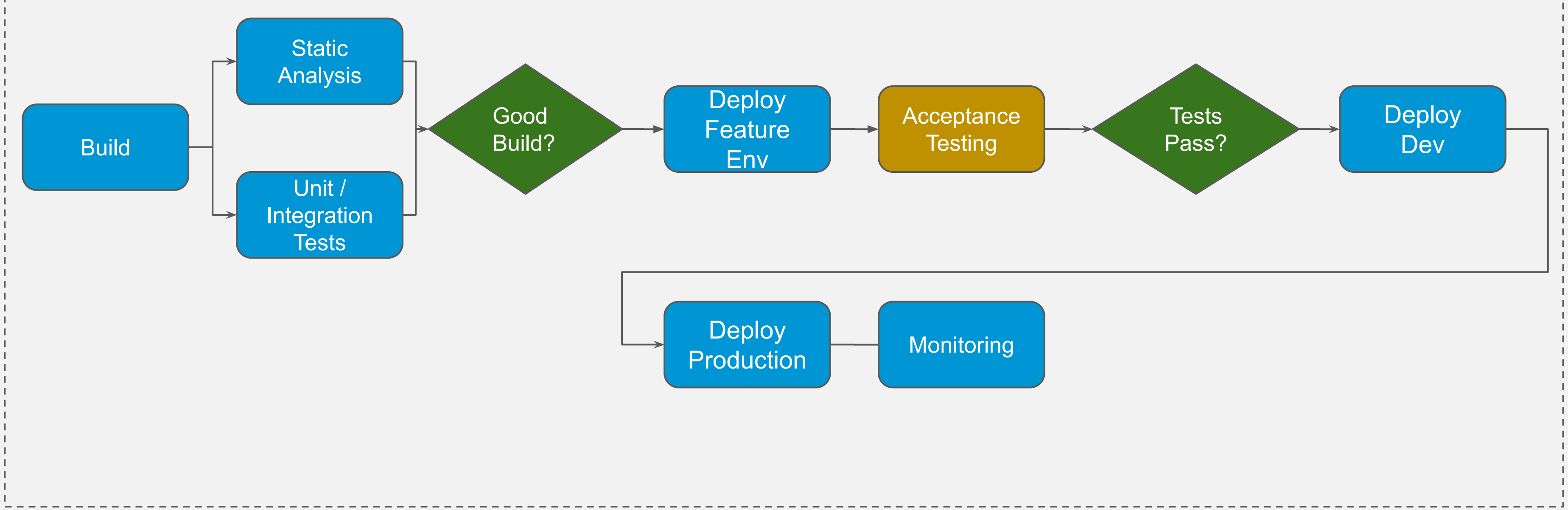
Retreating with Faux Safety	**Advancing with Real Safety** @JoshuaKerievsky
Decrease # of Releases	Increase # of Releases
Deploy once a month	Deploy continuously
Increase Sprint duration	Target continuous flow
Senior Architect reviews code changes	Create environment to continuously review changes
Control the integration of code	Whole team continuously integrates
Create dev, test and prod environments	Deploy from dev to production
Isolate work in long-lived branches	Perform trunk-based development
Perform manual testing	Automate checks & perform exploraory testing
Increase team size	Decrease team size

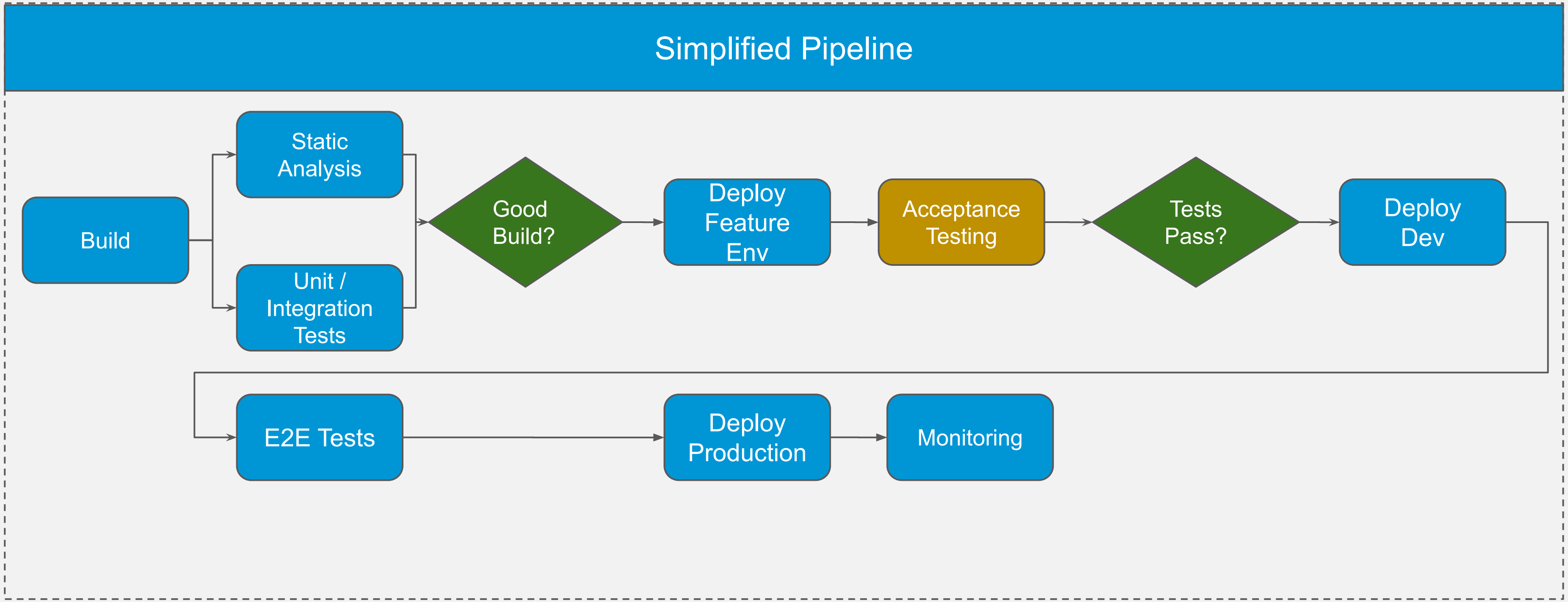
Don’t automate manual regression tests

Gaining trust

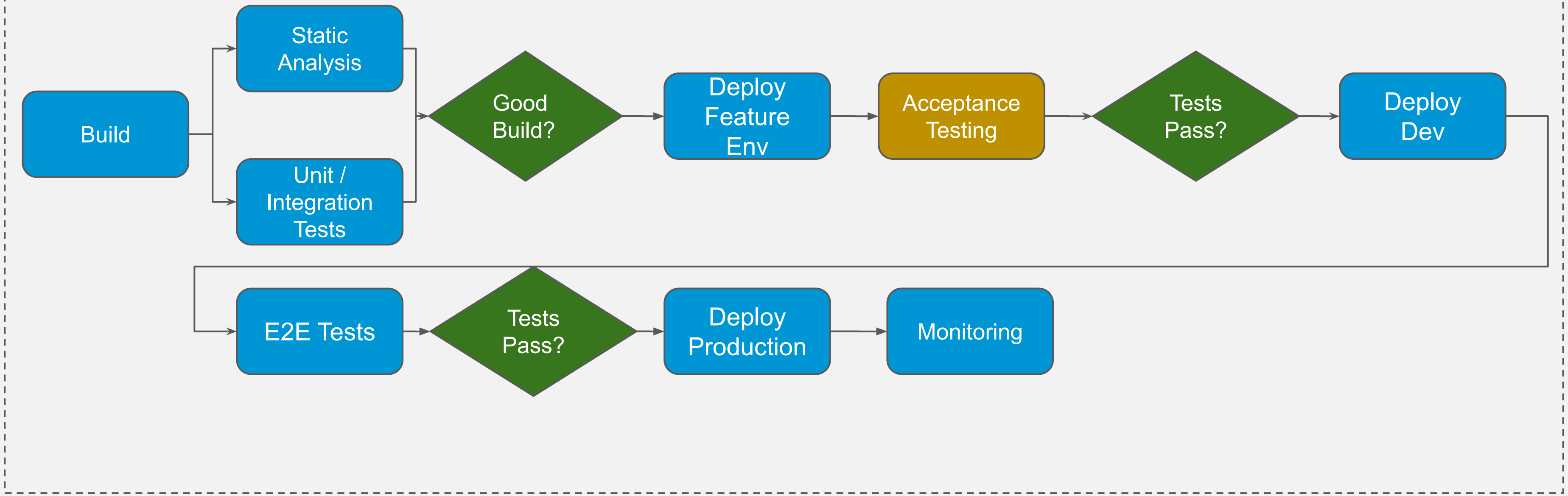
Photo by Marek Piwnicki on Unsplash

Simplified Pipeline



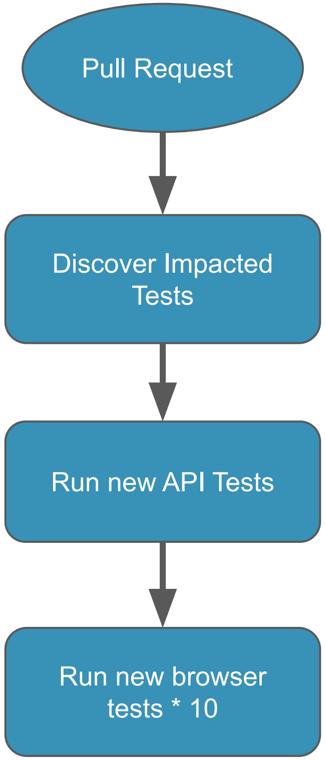

Simplified Pipeline
Build
Static Analysis
Unit / Integration Tests
Good Build?
Deploy Feature Env
Acceptance Testing
Tests Pass?
Deploy Dev
E2E Tests
Deploy Production
Monitoring

Simplified Pipeline

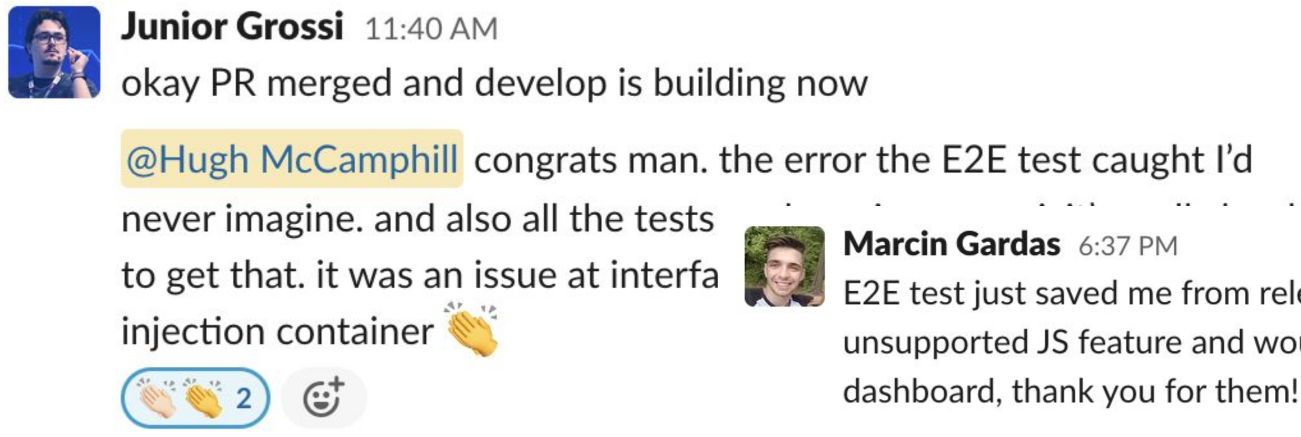


Maintaining trust

Test your tests



And wait….



Junior Grossi 11:40 AM

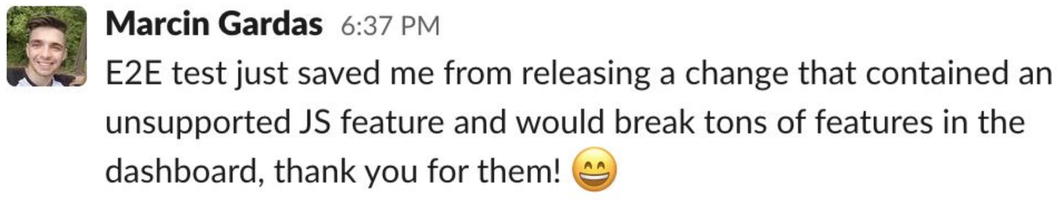
okay PR merged and develop is building now

@Hugh McCamphill congrats man. the error the E2E test caught I'd never imagine. and also all the tests

to get that. it was an issue at interfa

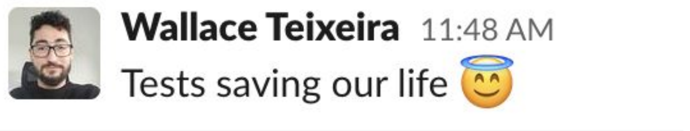
injection container 👏

👏👏 2



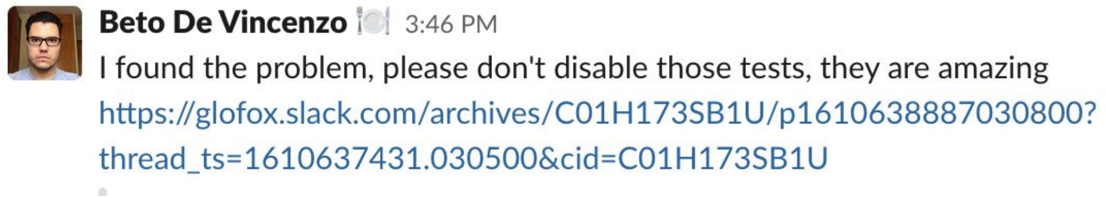
Marcin Gardas 6:37 PM

E2E test just saved me from releasing a change that contained an unsupported JS feature and would break tons of features in the dashboard, thank you for them! 😄



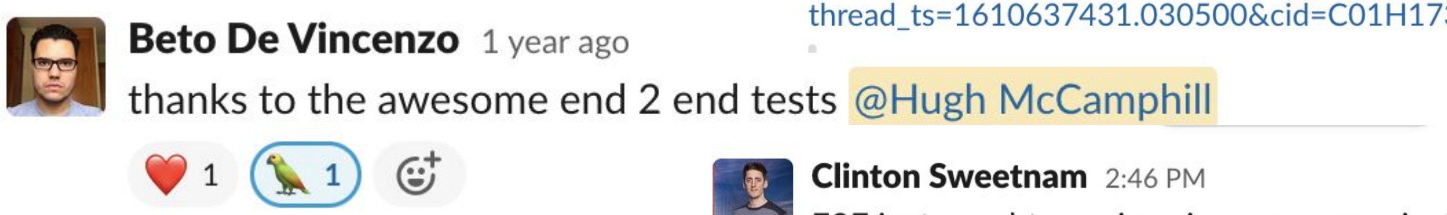
Wallace Teixeira 11:48 AM

Tests saving our life 😇



Beto De Vincenzo 🍽️ 3:46 PM

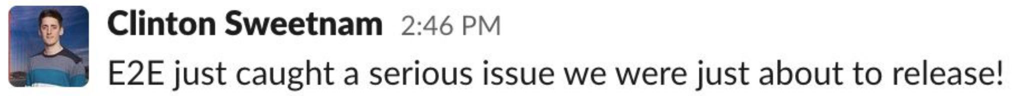
I found the problem, please don't disable those tests, they are amazing https://glofox.slack.com/archives/C01H173SB1U/p1610638887030800?thread_ts=1610637431.030500&cid=C01H173SB1U



Beto De Vincenzo 1 year ago

thanks to the awesome end 2 end tests @Hugh McCamphill

❤️ 1 🦜 1

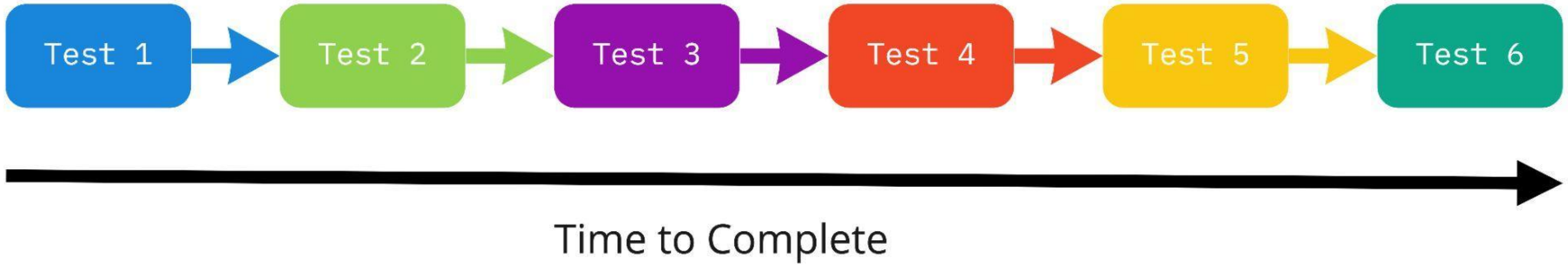


Clinton Sweetnam 2:46 PM

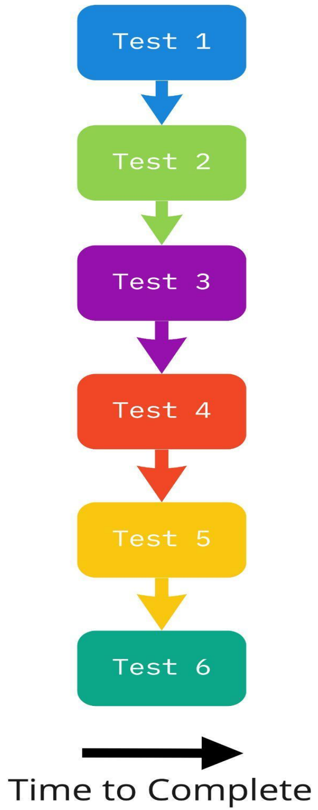
E2E just caught a serious issue we were just about to release!

Keep the run times short (ideally less than 10 minutes)

Running tests sequentially



Run tests in parallel



- Libraries and frameworks to support running in parallel
- Enough credentials to login in with
- Safe manipulation of shared data
- Create data as necessary

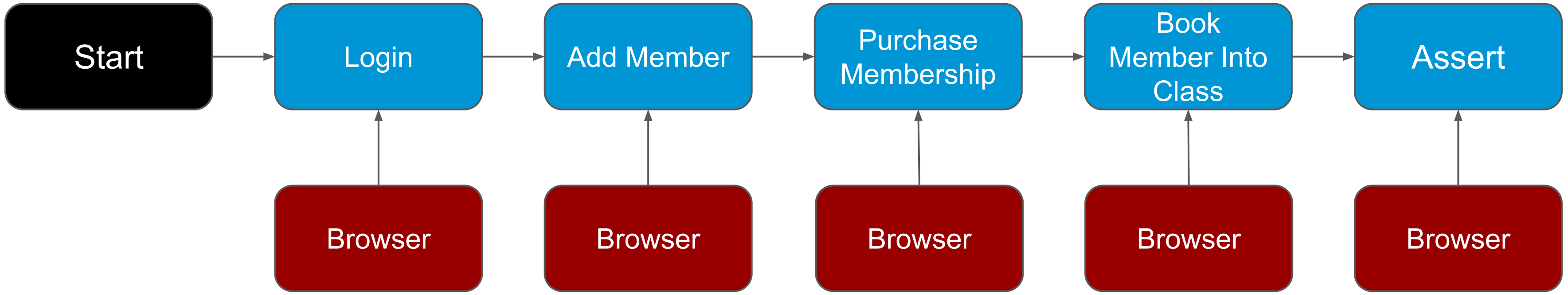
- Running the tests in multiple machine processes
- Running the tests across multiple (virtual) machines, eg – shard tests in CI

https://www.artstation.com/artwork/r64nO

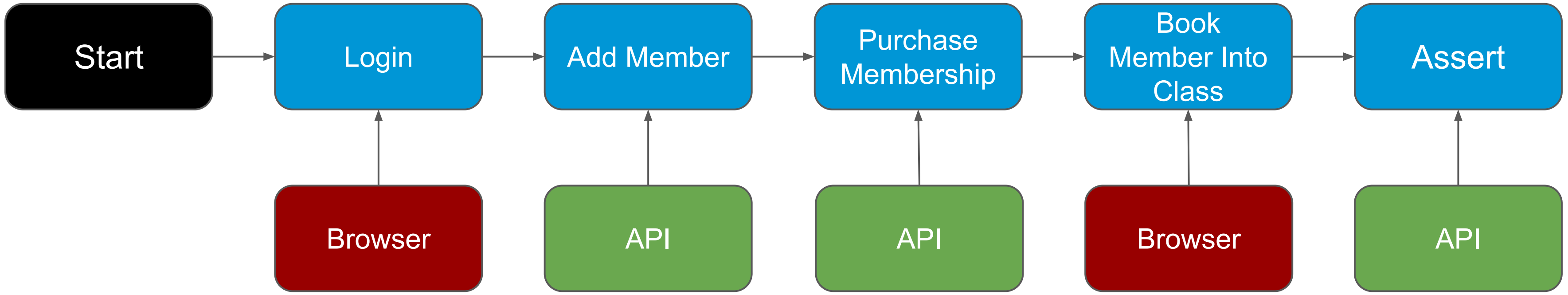
…benchmark data for 2020 again confirms that **longer tests directly lead to poor test quality**, as tests that complete in two minutes or less are **nearly twice** as likely to pass

Sauce Labs Continuous Testing Benchmark Report

Browser Driven Test



Using API for Setup and Assertion



```
describe('Demo', () => {
  it('Should be able to book member into class', async () => {
    await Dashboard.login();

    await Classes.addClassTomorrowForSinglePriceClients({ className });
    await Membership.createSinglePaymentMembershipPlan({ membershipName, planName });

    const member = new Member();
    await Client.addClient({ member });
    await Client.purchaseMembership({ membershipName, planName });

    await Dashboard.bookClass({ member, className });

    await Dashboard.openClient({ member });
    await ClientToolbarComponent.clickTransactions();

    const transaction = await Transactions.getTransaction(`${className} booking`);
    await expect(transaction.amount).toHaveText('$10.00');
  });
});
```

```
describe('Demo', () => {
  it('Should be able to book member into class', async () => {
    await Dashboard.login();

    await Classes.addClassTomorrowForSinglePriceClients({ className });
    await Membership.createSinglePaymentMembershipPlan({ membershipName, planName });

    const member = new Member();
    await Client.addClient({ member });
    await Client.purchaseMembership({ membershipName, planName });

    await Dashboard.bookClass({ member, className });

    await Dashboard.openClient({ member });
    await ClientToolbarComponent.clickTransactions();

    const transaction = await Transactions.getTransaction(`${className} booking`);
    await expect(transaction.amount).toHaveText('$10.00');
  });
});
```

BROWSER

BROWSER

```
describe('Demo', () => {
  it('Should be able to book member into class', async () => {
    await Dashboard.login();

    await new Class(className).create();
    const membershipService = await new MembershipService().create();

    const member = new Member();
    await member.create().purchaseMembershipService(membershipService);
    await member.membership('ACTIVE');

    await Dashboard.bookClass({ member, className });

    const transaction = await member.transaction('ENTITY_BOOKED');
    expect(transaction.event_context.invoice_amount).toEqual(10);
  });
});
```

```
describe('Demo', () => {
  it('Should be able to book member into class', async () => {
    await Dashboard.login();

    await new Class(className).create();
    const membershipService = await new MembershipService().create();

    const member = new Member();
    await member.create().purchaseMembershipService(membershipService);
    await member.membership('ACTIVE');
```

API

```
    await Dashboard.bookClass({ member, className });

    const transaction = await member.transaction('ENTITY_BOOKED');
    expect(transaction.event_context.invoice_amount).toEqual(10);
  });
});
```

API

Demo

Maintaining Tests

Photo by Markus Spiske on Unsplash

```
Do a GET Request and validate the response code and response body
    [documentation]  This test case verifies that the response code of the GET Request should be 
200,
    ...  the response body contains the 'title' key with value as 'London',
    ...  and the response body contains the key 'location_type'.
    [tags]  Smoke
    Create Session  mysession  https://www.metaweather.com  verify=true
    ${response}=  GET On Session  mysession  /api/location/search/  params=query=london
    Status Should Be  200  ${response}  #Check Status as 200

    #Check Title as London from Response Body
    ${title}=  Get Value From Json  ${response.json()}[0]  title
    ${titleFromList}=  Get From List   ${title}  0
    Should be equal  ${titleFromList}  London

    #Check location_type is present in the response body
    ${body}=  Convert To String  ${response.content}
    Should Contain  ${body}  location_type
```

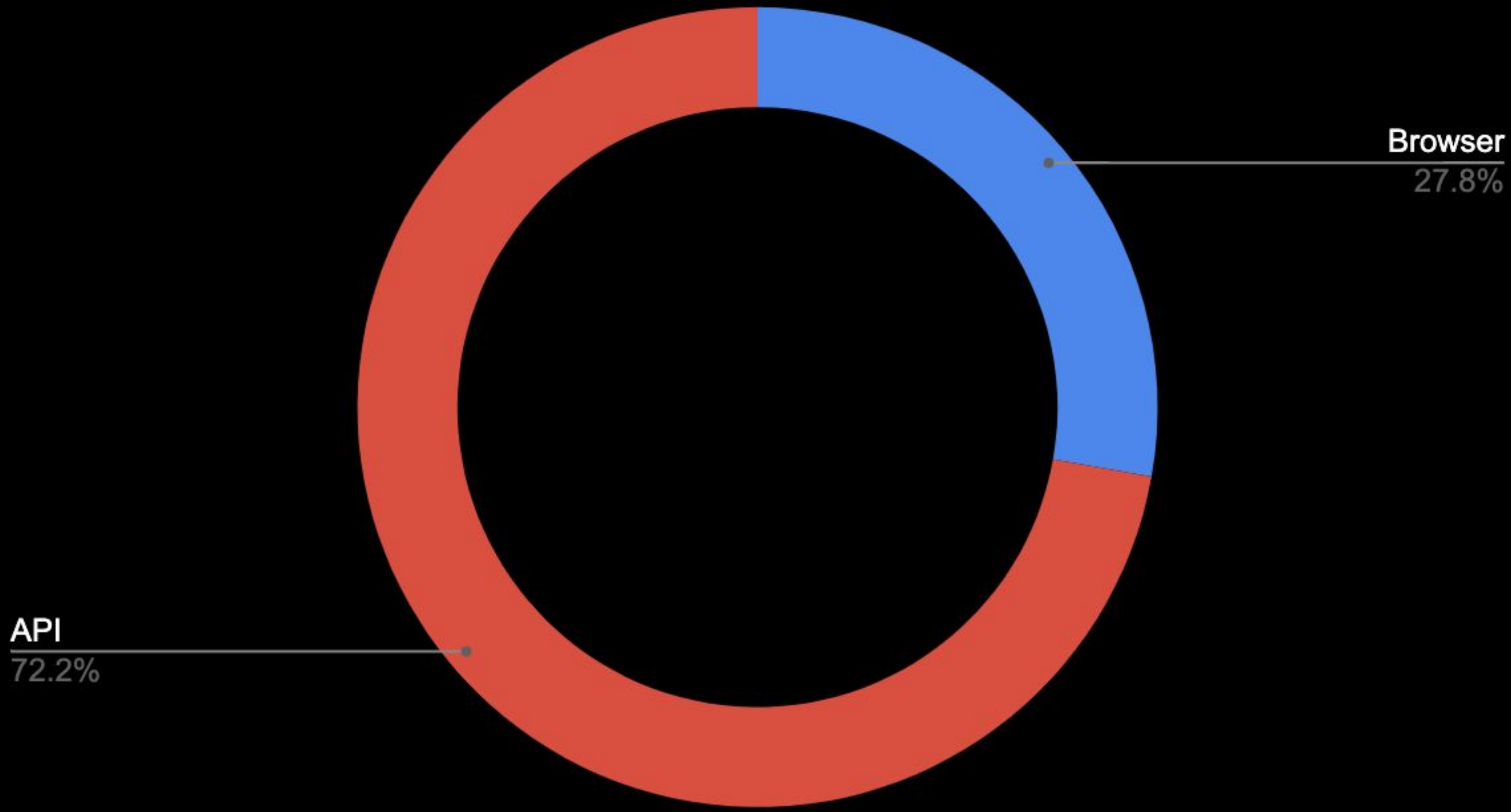
```
@Tag
Feature: Perform full demographic verification of a person

  Background:
    Given caller presents a valid access token

  Scenario: Perform full demographic verification with valid credentials
    Given a person with full demographic details
      | adhar_id      | 123456789     |
      | full_name     | John Doe      |
      | dob           | 31/12/1990    |
      | phone_no      | 999999999999  |
      | email         | john@doe.com  |
    And the match threshold is set to 1.0
    When request is submitted for full demographic verification
    Then verify that the HTTP response is 200
    And a transaction id is returned
```

```
describe('Demo', () => {
  it('Should be able to book member into class', async () => {
    await defaultStudio().superAdmin.login();

    await new Class(className).create();
    const membershipService = await new MembershipService().create();

    const member = new Member();
    await member.create().purchaseMembershipService(membershipService);
    await member.membership('ACTIVE');

    await member.bookClass(className);

    const transaction = await member.transaction('ENTITY_BOOKED');
    expect(transaction.event_context.invoice_amount).toEqual(10);
  });
});
```

Focus on the what not the how for better abstractions

How	**What**
Gets / Posts / Puts / Deletes	Page / Business objects
Clicking buttons	Domain Objects
Entering text	Screenplay pattern
Selecting from drop downs	

900 Tests



17

Thanks!