Agents, embeddings & fine-tuning, how we automate E2E-testing using AI

2024-11-12

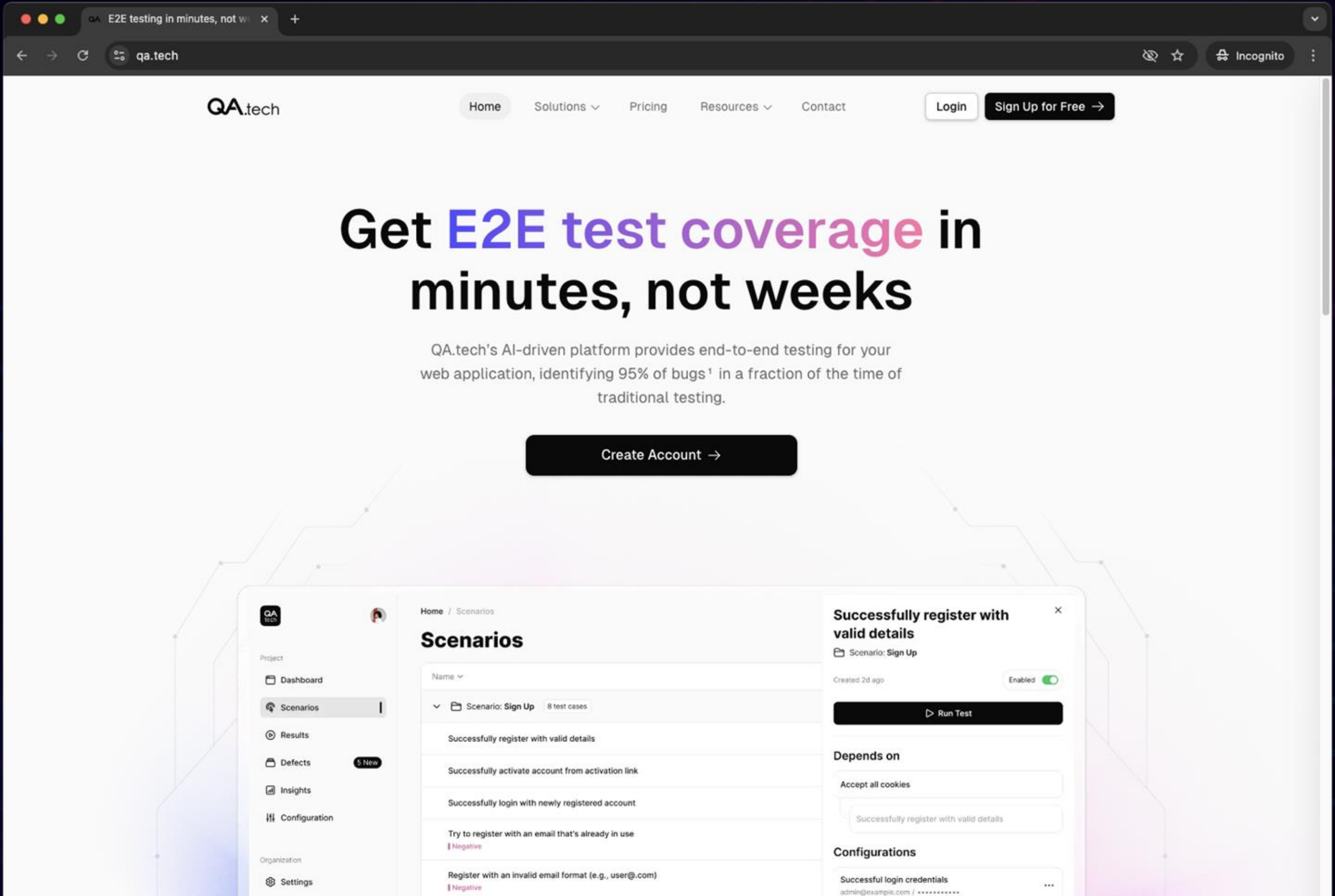
The Team

Daniel Mauno Pettersson
CEO / CTO

Tech leader and hands-on engineer for the last 15 years Previously CTO at memmo, Billogram, Dooer, CEO at Agigen

Vilhelm von Ehrenheim
CAIO

Built EQT's Motherbrain over the last 7 years,
Previously Data Science at Klarna



Get E2E test coverage in minutes, not weeks

QA.tech's AI-driven platform provides end-to-end testing for your web application, identifying 95% of bugs[1] in a fraction of the time of traditional testing.

Scenarios

- Successfully register with valid details
- Successfully activate account from activation link
- Successfully login with newly registered account
- Try to register with an email that's already in use
  Negative
- Register with an invalid email format (e.g., user@.com)
  Negative

Successfully register with valid details

Scenario: Sign Up

Created 2d ago

Depends on

Accept all cookies

Successfully register with valid details

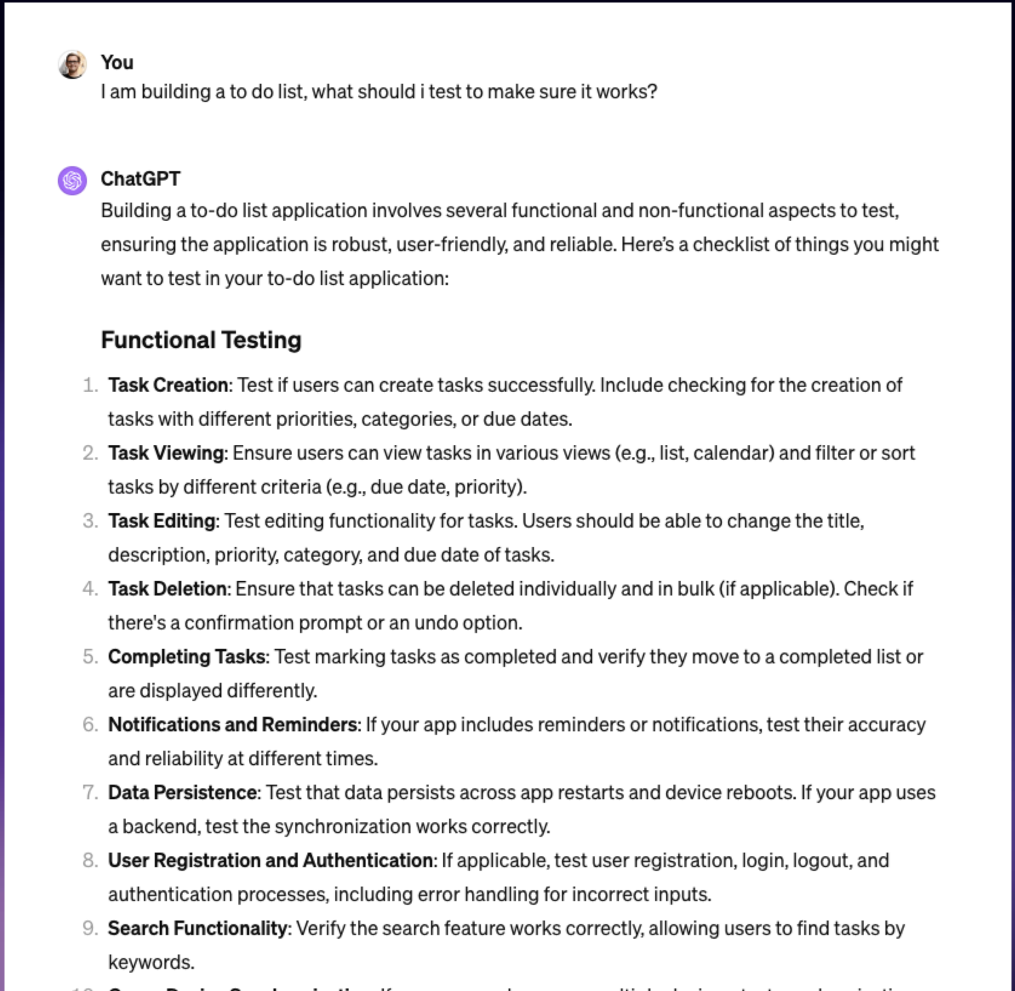
Configurations

Successful login credentials

admin@example.com / •••••••••••

Testing is Hard

- Today - products depend on many moving parts - **not only the code**
- Search tools, CMS, regional settings, A/B-testing, etc
- All affects user experience and performance
- Experienced this first hand as CTO at a B2C Marketplace



You

I am building a to do list, what should i test to make sure it works?

ChatGPT

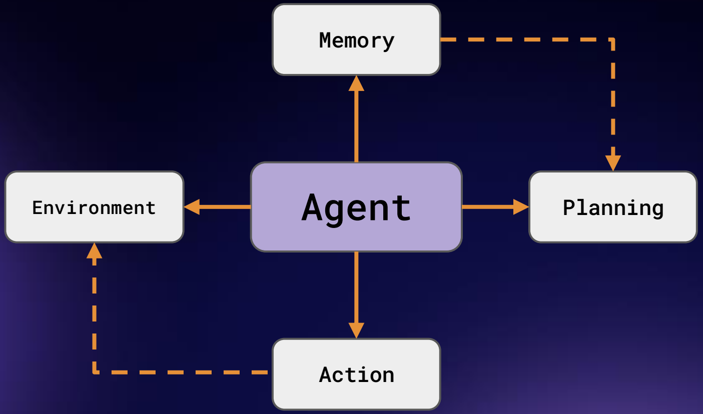
Building a to-do list application involves several functional and non-functional aspects to test, ensuring the application is robust, user-friendly, and reliable. Here's a checklist of things you might want to test in your to-do list application:

Functional Testing

1. **Task Creation**: Test if users can create tasks successfully. Include checking for the creation of tasks with different priorities, categories, or due dates.
2. **Task Viewing**: Ensure users can view tasks in various views (e.g., list, calendar) and filter or sort tasks by different criteria (e.g., due date, priority).
3. **Task Editing**: Test editing functionality for tasks. Users should be able to change the title, description, priority, category, and due date of tasks.
4. **Task Deletion**: Ensure that tasks can be deleted individually and in bulk (if applicable). Check if there's a confirmation prompt or an undo option.
5. **Completing Tasks**: Test marking tasks as completed and verify they move to a completed list or are displayed differently.
6. **Notifications and Reminders**: If your app includes reminders or notifications, test their accuracy and reliability at different times.
7. **Data Persistence**: Test that data persists across app restarts and device reboots. If your app uses a backend, test the synchronization works correctly.
8. **User Registration and Authentication**: If applicable, test user registration, login, logout, and authentication processes, including error handling for incorrect inputs.
9. **Search Functionality**: Verify the search feature works correctly, allowing users to find tasks by keywords.

What are AI Agents?

- An **AI agent** is an AI that can **interact** with an **environment** autonomously.
- Components of an agent are typically **planning**, **memory**, **actions** & **observation.**
- **Example:** An AI assistant that can book you a table at a restaurant.



Why AI Agents Matter

- Up until now code has always been **deterministic** - meaning always doing the same thing
- You needed to know **what input to expect** before you build it
- Agents allow programs to **make decisions autonomously** on your behalf
- **We will be able to do a lot more tasks, that previously required a human in the loop**

Why Testing with Agents?

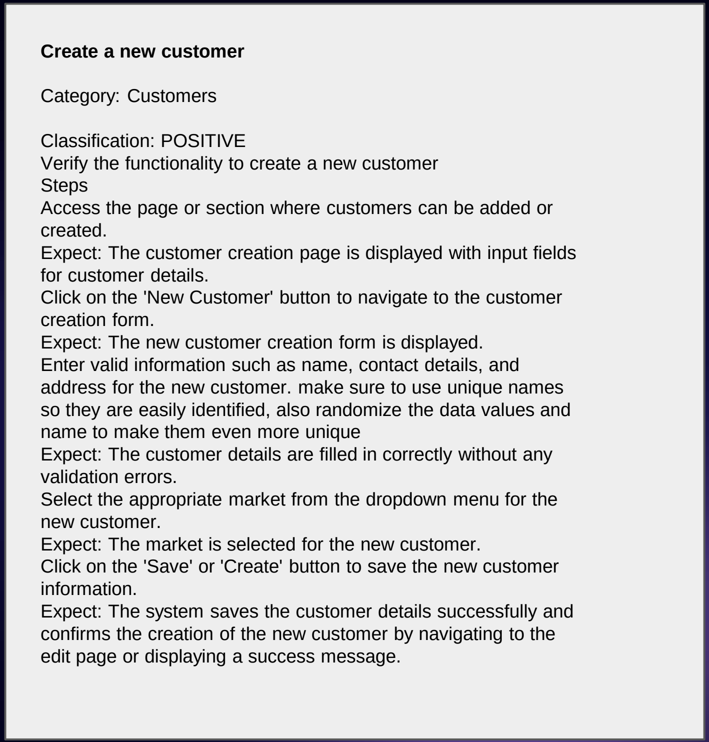
All web applications needs to be tested before release and the process is slow and expensive, stealing time from building new features

- We are already **not testing enough**
- # of new features, # of releases, will increase as developers **get more productive** using AI tools
- Current test automation tools are **not handling changes** well
- Manual testing is too **time consuming**

How QA.tech Approaches Testing

All web applications needs to be tested before release and the process is slow and expensive, stealing time from building new features

- Tests are written in **natural language**
- Can **act and evaluate results**, as a human would
- Available **24/7**, knows every spoken language in the world, can scale with the needs of the organization



Create a new customer

Category: Customers

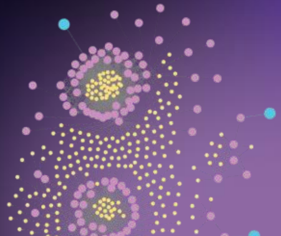
Classification: POSITIVE
Verify the functionality to create a new customer
Steps
Access the page or section where customers can be added or created.
Expect: The customer creation page is displayed with input fields for customer details.
Click on the 'New Customer' button to navigate to the customer creation form.
Expect: The new customer creation form is displayed.
Enter valid information such as name, contact details, and address for the new customer. make sure to use unique names so they are easily identified, also randomize the data values and name to make them even more unique
Expect: The customer details are filled in correctly without any validation errors.
Select the appropriate market from the dropdown menu for the new customer.
Expect: The market is selected for the new customer.
Click on the 'Save' or 'Create' button to save the new customer information.
Expect: The system saves the customer details successfully and confirms the creation of the new customer by navigating to the edit page or displaying a success message.

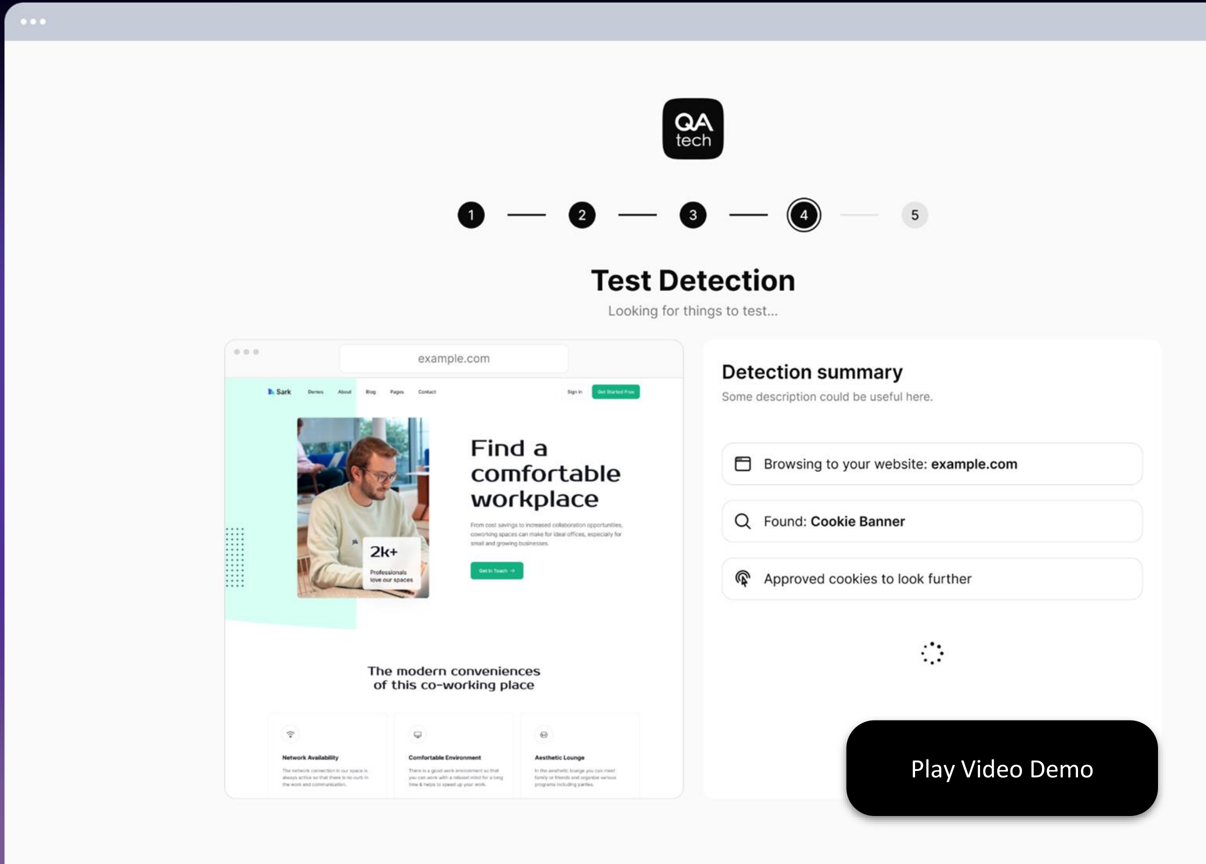
Product - Discover

We only need a URL to your website to discover and detect tests for your product

Analysis & Test Generation

- Map page functionality
- Suggest tests for discovered user flows
- Compute coverage analysis





Test Detection

Looking for things to test...

Detection summary

Some description could be useful here.

Browsing to your website: **example.com**

Found: **Cookie Banner**

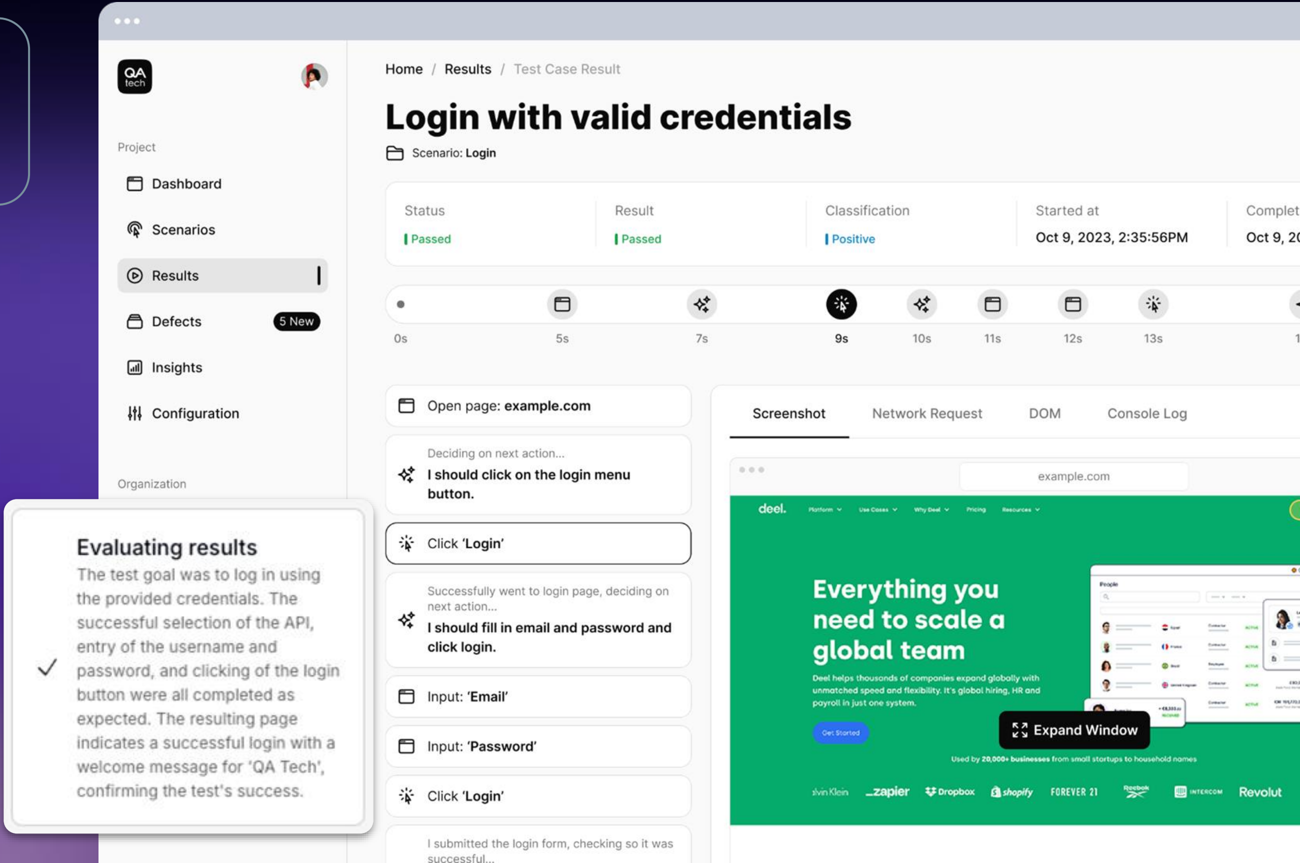
Approved cookies to look further

Product - Run tests

Triggers automatically as part of the development cycle and provides direct feedback to the developer.

Perform the objective

- The agent starts and operates a browser
- Agent attempts to achieve the objective
- Interact with the page as a human
- Dynamically evaluate outcome against objective



Home / Results / Test Case Result

Login with valid credentials

Scenario: **Login**

Status	Result	Classification	Started at	Complet
Passed	Passed	Positive	Oct 9, 2023, 2:35:56PM	Oct 9, 20

Open page: **example.com**

Deciding on next action...
I should click on the login menu button.

Click '**Login**'

Successfully went to login page, deciding on next action...
I should fill in email and password and click login.

Input: '**Email**'

Input: '**Password**'

Click '**Login**'

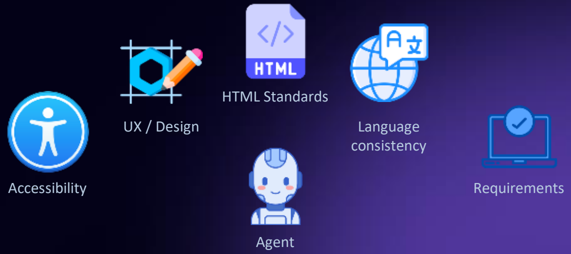
I submitted the login form, checking so it was successful...

Evaluating results

✓ The test goal was to log in using the provided credentials. The successful selection of the API, entry of the username and password, and clicking of the login button were all completed as expected. The resulting page indicates a successful login with a welcome message for 'QA Tech', confirming the test's success.

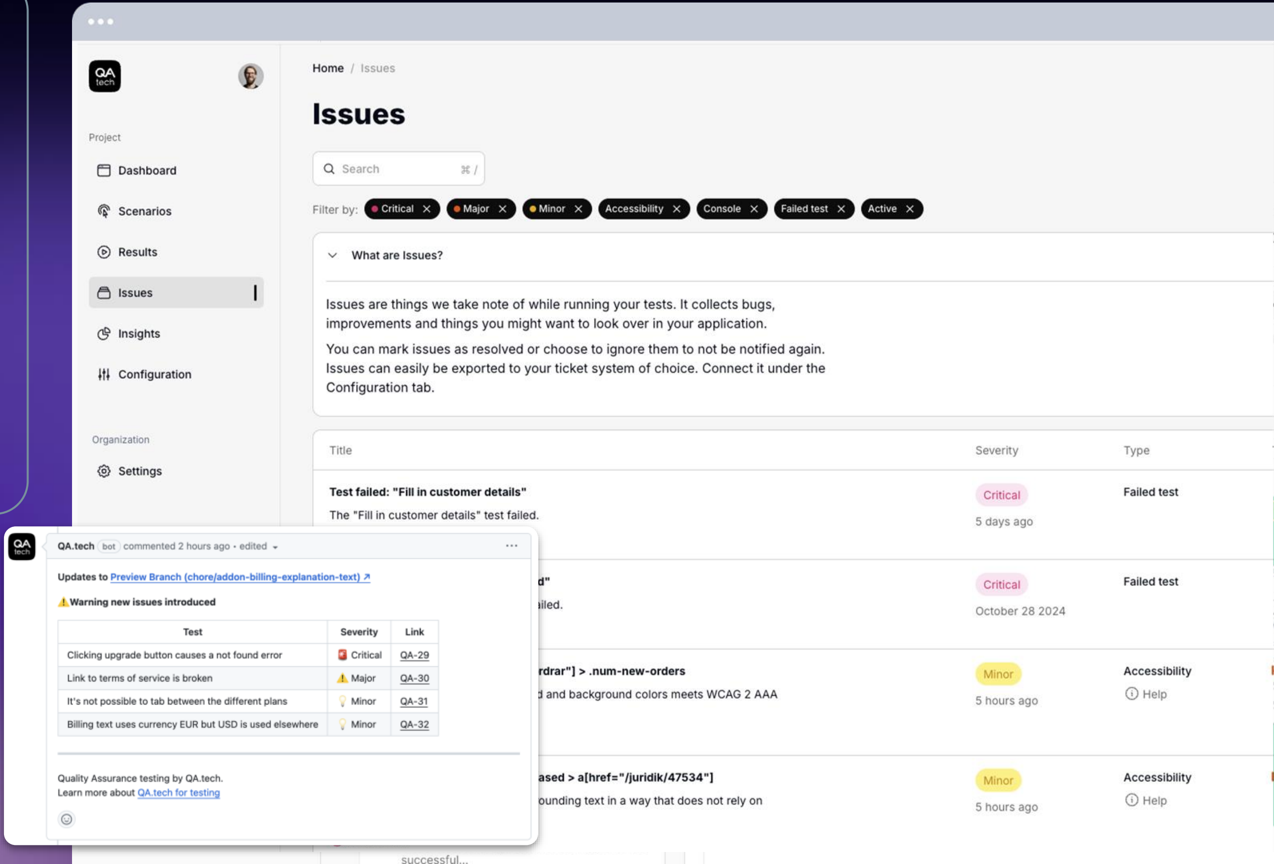
Product - Analyze session

Session analysis



Multiple AIs reviews each agents session and report issues

Issues are identified and packaged into developer friendly bug reports.



Issues

What are Issues?

Issues are things we take note of while running your tests. It collects bugs, improvements and things you might want to look over in your application.

You can mark issues as resolved or choose to ignore them to not be notified again. Issues can easily be exported to your ticket system of choice. Connect it under the Configuration tab.

Title	Severity	Type
Test failed: "Fill in customer details" The "Fill in customer details" test failed.	Critical 5 days ago	Failed test
d" iled.	Critical October 28 2024	Failed test
rdrar"] > .num-new-orders d and background colors meets WCAG 2 AAA	Minor 5 hours ago	Accessibility ⓘ Help
ased > a[href="/juridik/47534"] ounding text in a way that does not rely on	Minor 5 hours ago	Accessibility ⓘ Help
successful...		

QA.tech bot commented 2 hours ago • edited

Updates to Preview Branch (chore/addon-billing-explanation-text) ↗

⚠️Warning new issues introduced

Test	Severity	Link
Clicking upgrade button causes a not found error	🚨 Critical	QA-29
Link to terms of service is broken	⚠️ Major	QA-30
It's not possible to tab between the different plans	💡 Minor	QA-31
Billing text uses currency EUR but USD is used elsewhere	💡 Minor	QA-32

Quality Assurance testing by QA.tech.
Learn more about QA.tech for testing

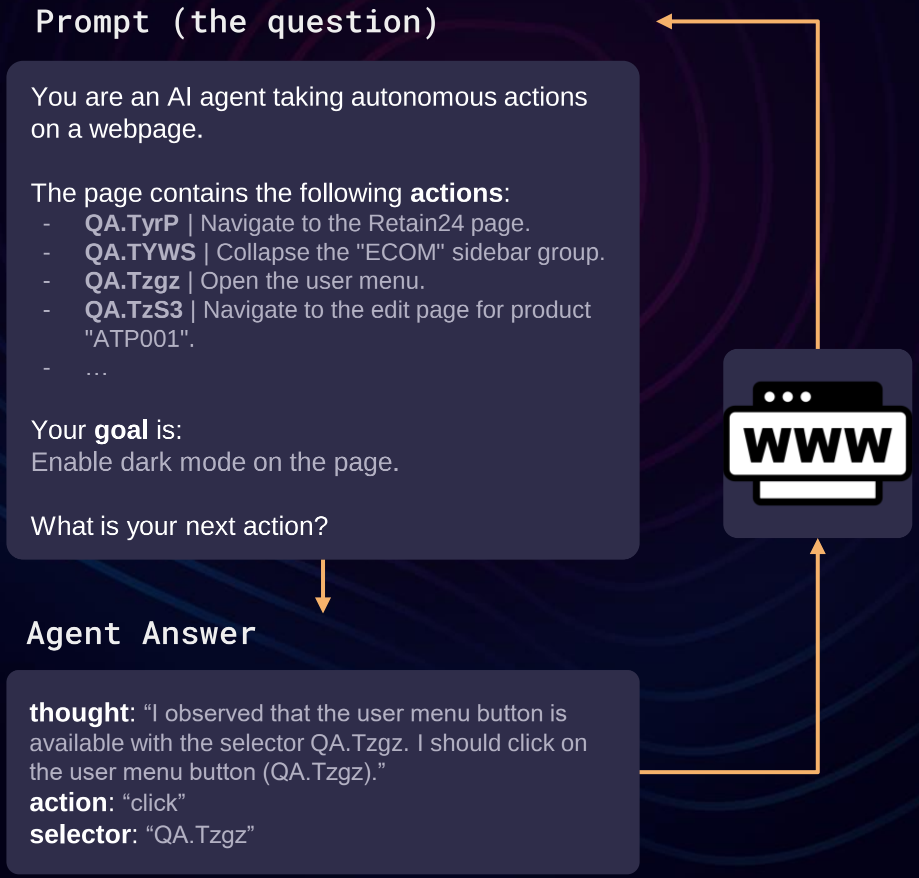
Deep Dive to AI Agents

- **The AI agent** - How an AI agent functions
- **Knowledge Graph** - Learning from interactions
- **Fine-Tuning** - Continuous model improvements

Vilhelm von Ehrenheim
Co-Founder & Chief AI Officer @ QA.tech

Deepdive: **AI Agent**

- An AI agent can be just a **question** run in a **loop**, over and over.
- Each loop asks the agent to select an **action** and then provides **feedback** from the environment
- LLMs can respond in structured format such as JSON to simplify integration.



`Prompt (the question)`

You are an AI agent taking autonomous actions on a webpage.

The page contains the following **actions**:

- **QA.TyrP** | Navigate to the Retain24 page.
- **QA.TYWS** | Collapse the "ECOM" sidebar group.
- **QA.Tzgz** | Open the user menu.
- **QA.TzS3** | Navigate to the edit page for product "ATP001".
- …

Your **goal** is:
Enable dark mode on the page.

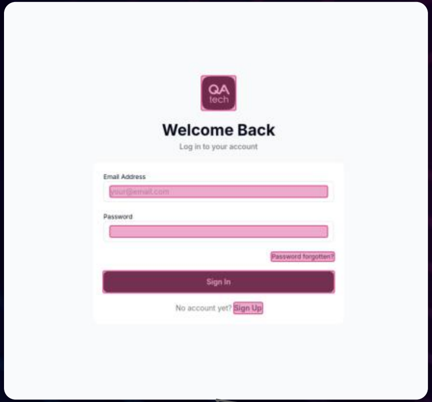
What is your next action?

`Agent Answer`

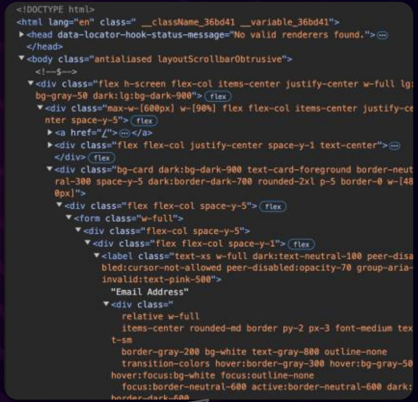
thought: “I observed that the user menu button is available with the selector QA.Tzgz. I should click on the user menu button (QA.Tzgz).”
action: “click”
selector: “QA.Tzgz”

Deepdive: **AI Agent**

- To make the question clear, we **pre-process** the interface before we present it to the agent.
- We look at both the **HTML** and the **visual interface** to provide clear descriptions of actions.
- This helps us handle changes over time which is traditionally hard to handle in hardcoded tests.



+



```
<!DOCTYPE html>
<html lang="en" class=" __className_36bd41 __variable_36bd41">
 ▶<head data-locator-hook-status-message="No valid renderers found.">⋯
  </head>
 ▼<body class="antialiased layoutScrollbarObtrusive">
    <!--$-->
   ▼<div class="flex h-screen flex-col items-center justify-center w-full lg:
    bg-gray-50 dark:lg:bg-dark-900"> flex
     ▼<div class="max-w-[600px] w-[90%] flex flex-col items-center justify-ce
      nter space-y-5"> flex
       ▶<a href="/">⋯</a>
       ▶<div class="flex flex-col justify-center space-y-1 text-center">⋯
        </div> flex
       ▼<div class="bg-card dark:bg-dark-900 text-card-foreground border-neut
        ral-300 space-y-5 dark:border-dark-700 rounded-2xl p-5 border-0 w-[48
        0px]">
         ▼<div class="flex flex-col space-y-5"> flex
           ▼<form class="w-full">
             ▼<div class="flex-col space-y-5">
               ▼<div class="flex flex-col space-y-1"> flex
                 ▼<label class="text-xs w-full dark:text-neutral-100 peer-disa
                  bled:cursor-not-allowed peer-disabled:opacity-70 group-aria-
                  invalid:text-pink-500">
                    "Email Address"
                   ▼<div class="
                      relative w-full
                      items-center rounded-md border py-2 px-3 font-medium tex
                    t-sm
                      border-gray-200 bg-white text-gray-800 outline-none
                      transition-colors hover:border-gray-300 hover:bg-gray-50
                    hover:focus:bg-white focus:outline-none
                      focus:border-neutral-600 active:border-neutral-600 dark:
                    border-dark-600
```

AI Embedding

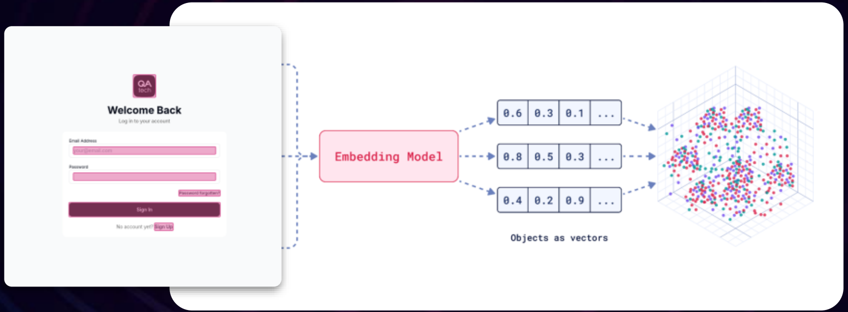
AI Description

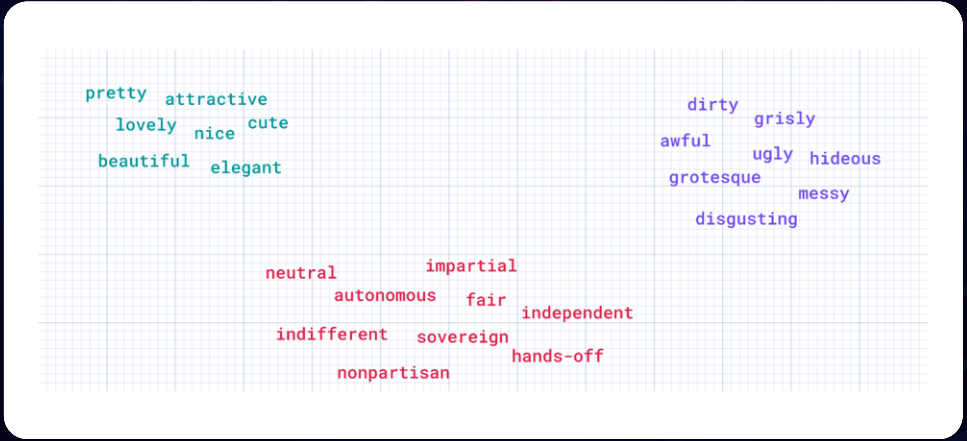
1. QA.T7XN - Navigate to the sign up page.
2. QA.TJ9w - Navigate to the password reset page.
3. QA.Tc4l - Navigate to the home page.
4. QA.T1Nh - Enter your email address.
5. QA.Tj1J - Enter your password.
6. QA.T8FJ - Submit the sign in form.

Deepdive: **AI Agent**

What are **embeddings**?

- An embedding (or vector) is a numerical **representation of meaning**.
- Represented as a point in a **high dimensional space**.
- Close points are “**similar**”.
- We use this to be **robust to changes** in selectors.

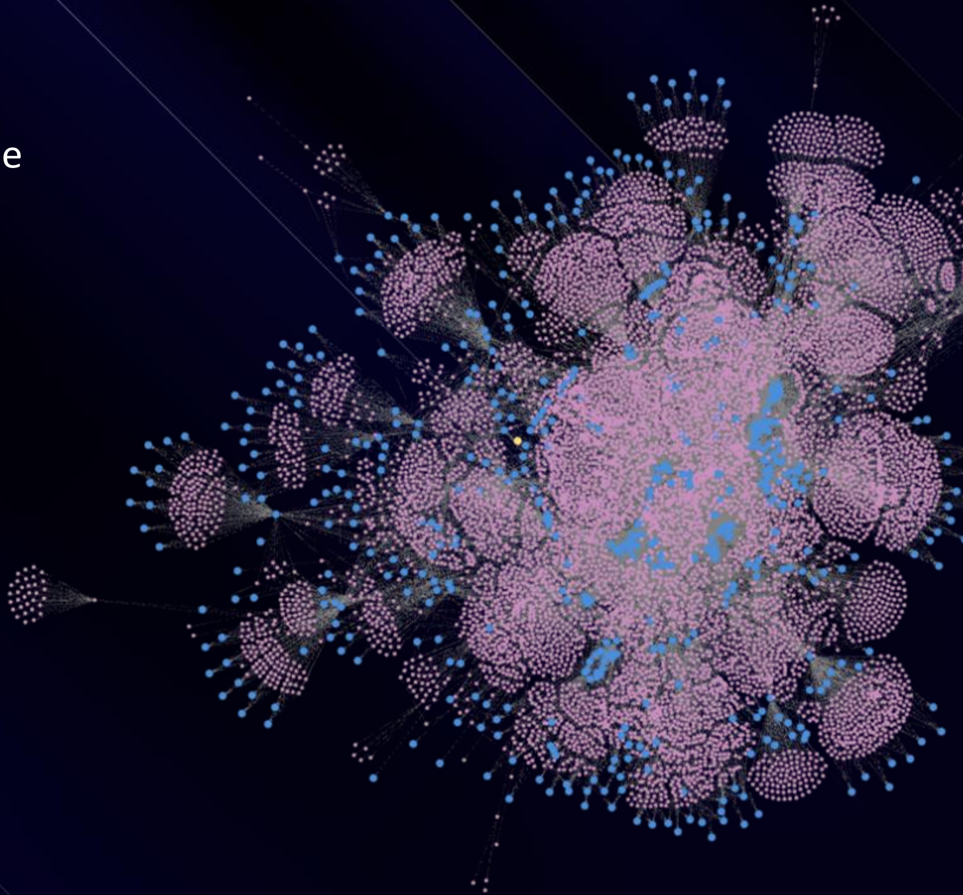




Deepdive: **Knowledge Graph**

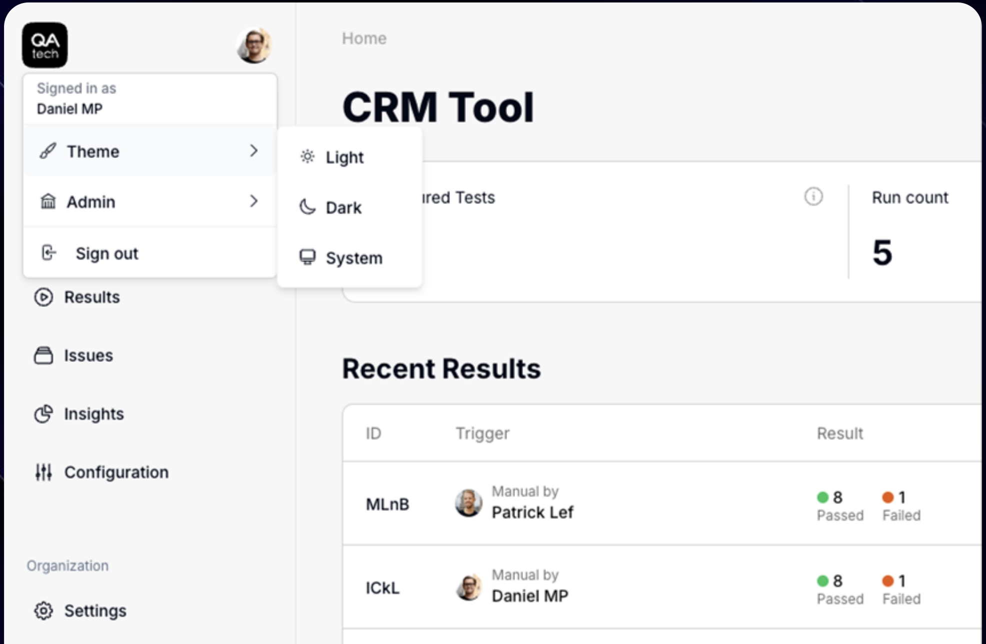
We map out the page into a **knowledge graph** to be able to understand the page

- Available actions
- Execution trajectories
- Page descriptions and embeddings



Deepdive: **Knowledge Graph**

What happens when you click a button?



CRM Tool

ıred Tests

Run count

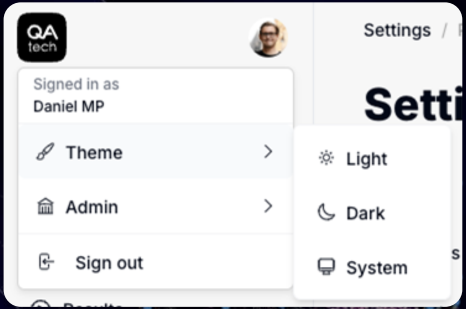
5

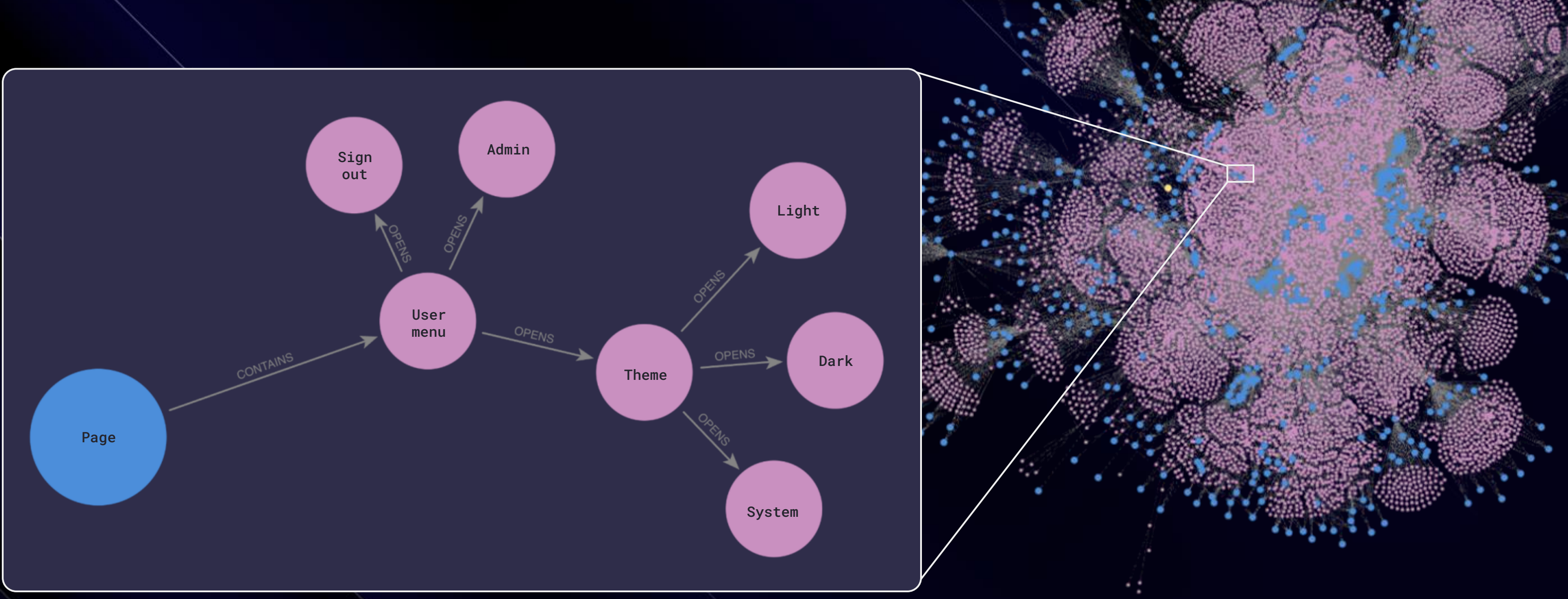
Recent Results

ID	Trigger	Result	
MLnB	Manual by Patrick Lef	● 8 Passed	● 1 Failed
ICkL	Manual by Daniel MP	● 8 Passed	● 1 Failed

Deepdive: **Knowledge Graph**

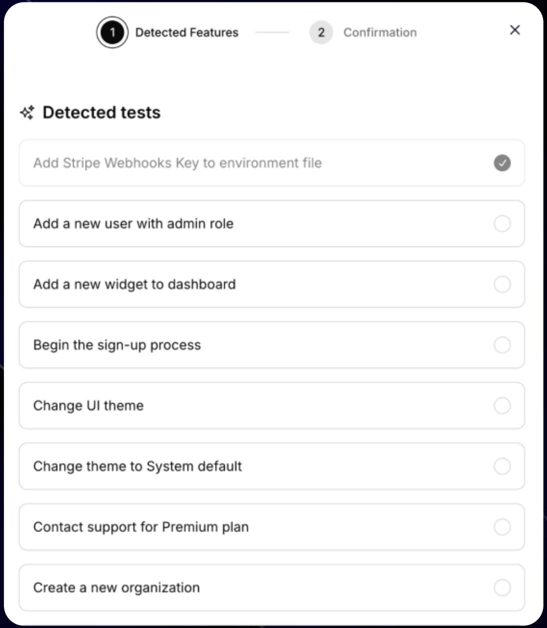
What happens when you click a button?





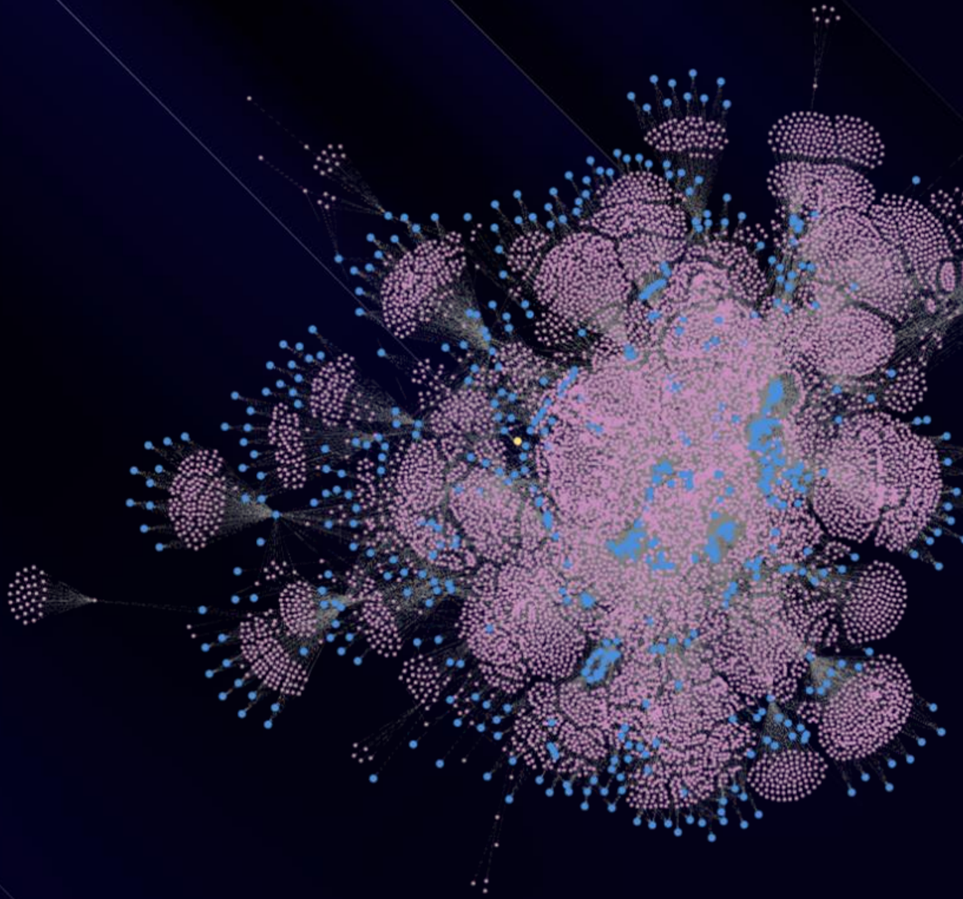
Deepdive: **Knowledge Graph**

What exists on a page that we can test?



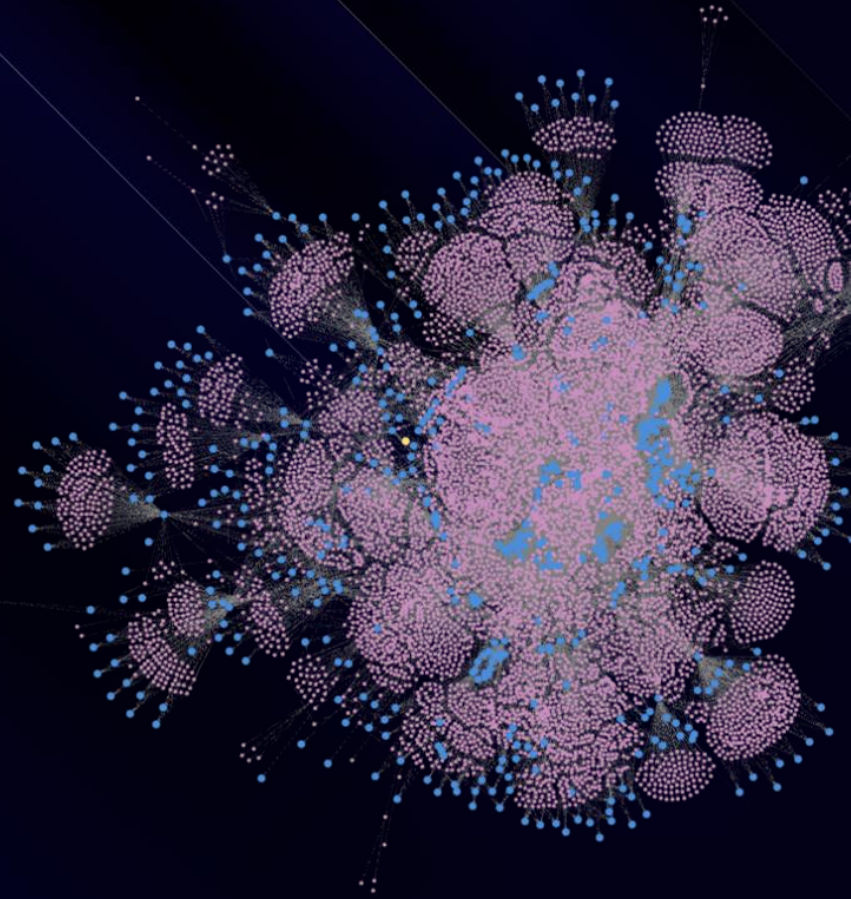
Detected tests

- Add Stripe Webhooks Key to environment file ☑
- Add a new user with admin role ○
- Add a new widget to dashboard ○
- Begin the sign-up process ○
- Change UI theme ○
- Change theme to System default ○
- Contact support for Premium plan ○
- Create a new organization ○



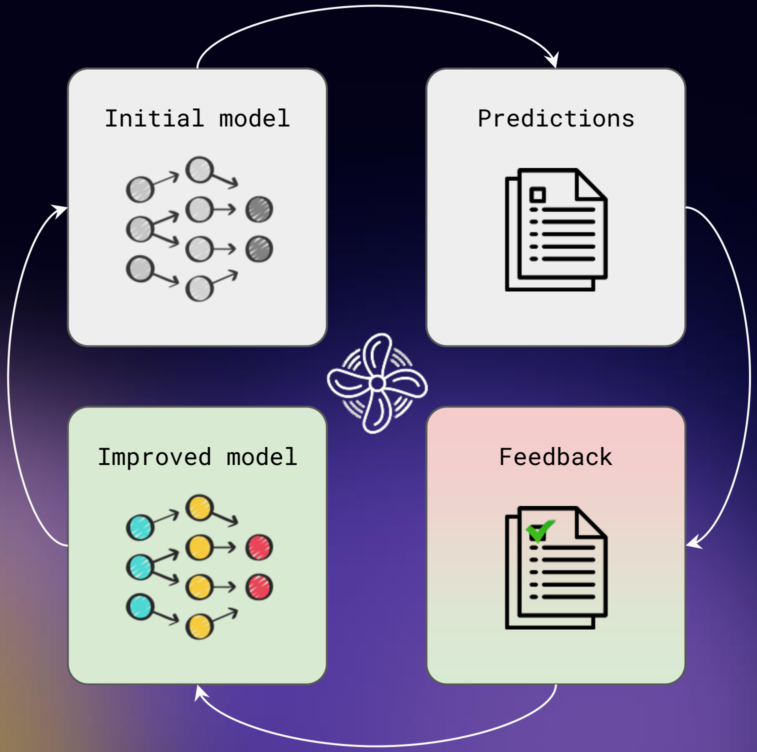
Deepdive: **Knowledge Graph**

We can also use the graph for **optimal route planning** for the agent



Deepdive: **Fine-Tuning**

- How can the agent improve with experience?
- **Fine-tuning** is the process of continuing training a model on new examples.
- Can be both human annotated (supervised) or based on patterns in the data (unsupervised).
- This flywheel greatly **improve performance** over time.



Deepdive: **Fine-Tuning**

The process of fine-tuning a model

Scope	Select	Train	Evaluate	Deploy
Define a task Build dataset	Select model candidate	Prompt engineering, fine-tuning	Define metrics Evaluate on hold-out data	Deploy model for inference Compression Collect human feedback

The Future

- AI will help humans **automate more common tasks**
- Humans still need to **guide and influence** the Agents
- Velocity of product development, and testing, will **increase**
- Just like unit testing **-** e2e testing will be **everywhere**
- Why would you let **your users** discover bugs?
  You can let **synthetic users** try everything!

Try QA.tech for free

Try it out today →

https://qa.tech

Thank You!

Any Questions?

Contact

Daniel Mauno Pettersson
daniel@qa.tech
+46 733 44 5315

Resources

Website:
https://qa.tech/