ARCH DEVOPS PRESENTS

How to Get Started With Automation

by creating your first project

What We're Doing

Software development and testing are in a really interesting place right now. Breakthroughs made in the last few years have resulted in a complexity boom.

Trillions of pieces of info are flying around by the second and testers are often the last bastion of quality. But, too often, that phase gets squished in favor of shipping quickly.

Arch DevOps realizes this looks like testing is becoming obsolete. But the fact is, the same technology advancements that create all that complexity have also helped create easy ways for testers to automate a lot of their work.

Not only that, but learning these skills increases their value and (let's be real here), their marketability.

The problem is, many people don't know where to start.

There are hundreds of ways to start on the path of learning automation. And speaking as a tester, we often think of the worst case scenarios, such as: "What if I learn the wrong thing?" or "What if I get stuck?"

But you'll never be learning the wrong thing. And as we'll see later: you aren't going to get stuck. At least, not permanently.

The purpose of this guide is to help you get started on the path.

Just beginning with something, and allowing your skill to grow organically, will get you on the path toward mastery sooner than you might think!

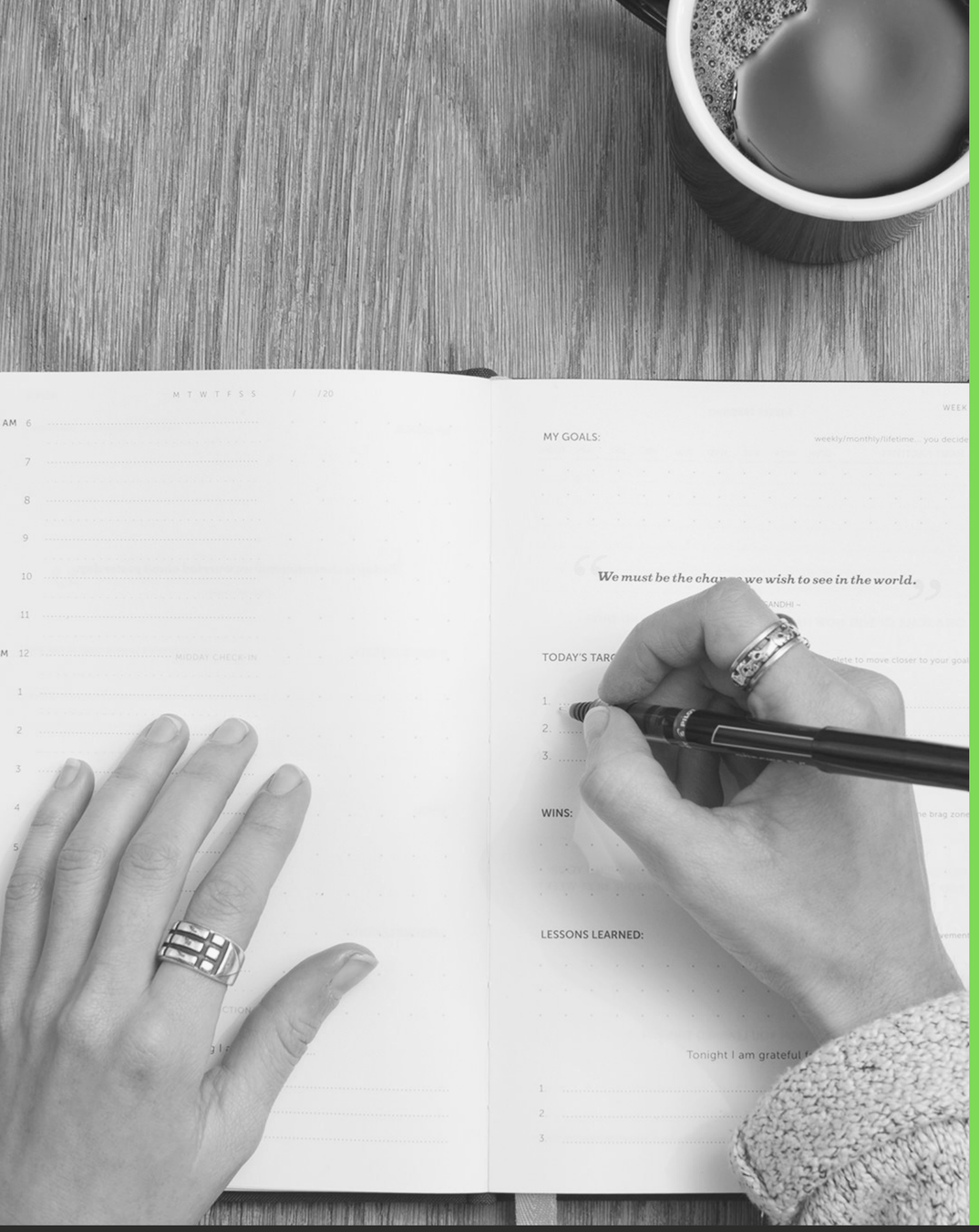
We'll focus on automating a browser based task, because this is the most common thing that people want to automate.

Not only is it something that can save a lot of time, but it's a little more fun since you can see the results of your work!

By the end of this guide you'll have completed your first small project:

Sending Fritz a message on LinkedIn

Looking forward to helping you on these first steps of your journey!
--Michael "Fritz" Fritzius



First: Mindset

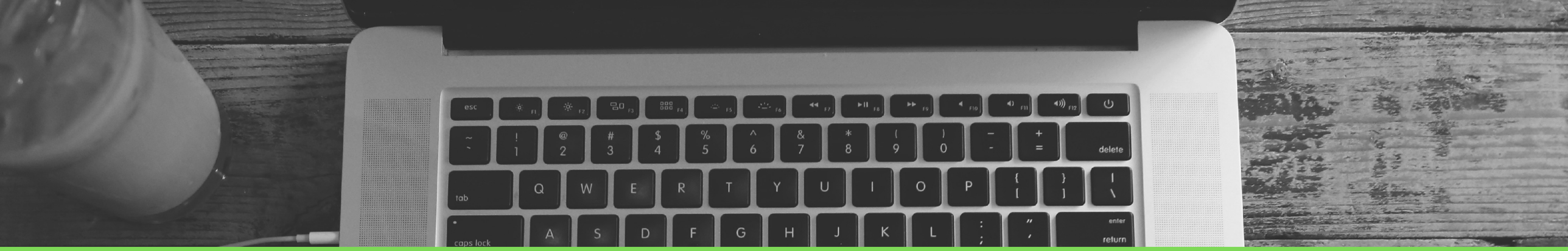
Don't boil the ocean. Automation doesn't have to be a huge endeavor. We're going to focus on exactly what we need to do to get started.

Overcome inertia. The hardest step is the first one, but then the steps get easier and the learning goes faster. Once you get going you'll find that you're grasping concepts quicker and figuring things out at lightning speed.

Anything can be automated. I've yet to get stuck permanently on automating anything. there's always a way to do it. so don't worry about running forward and getting stuck. worst case, it's just a hurdle that we have to jump over.

Decompose. Much of automation comes from breaking complex ideas into simple pieces. Instead of: "How do I send a message on LinkedIn?" think: "How do I click a button? Enter text into a field?" There are plenty of articles, posts and people that can help you figure out exactly how to do a specific thing with automation. That's why much of this guide will focus on concepts instead of code samples.

You've got this. Now let's get a place where we can play!



Get a Workspace

SUBLIME TEXT EDITOR

Sublime Text Editor is powerful text editor. It's also free (although it'll yell once in awhile about buying a copy).

I do support the tool, they do great work.

It's also open source so later if you want to write plugins for it to automate different tasks, that's an option too!

This will be where we build out automated scripts.

Download it from **https://www.sublimetext.com/** and install. It shouldn't take long.

SET UP THE STACK

1

RUBY

rubyinstaller.org

2

WATIR

watir.com

3

CHROMEDRIVER

chromedriver.chromium.org

SET UP THE STACK CONT'D

RUBY

Fairly easy language to pick up.

We don't want the language itself getting in the way of learning.

Plus it's easy to install and is open source.

Download the suggested version, which should be in BOLD on the download page.

WATIR

This is the Ruby gem used to drive a browser.

It handles interacting with the browser, clicking buttons, entering text, selecting from dropdowns, and a bunch of other things.

Good times are ahead.

CHROMEDRIVER

This is a "go-between" for your script, and the actual chrome browser.

It injects low-level commands into a browser instance that it brings up.

Grab one from the Chromedriver site and put it in your path.

NEXT STEP:

Opening the Browser

The watir website has a great example right on the front page of how to open a browser.

There's minimal setup required. In fact, you can probably copy and paste the example into your own script and run it.

Action item: Create a simple script using the example on the front page

GO TO LINKEDIN

First, if we're not connected already, go ahead and connect with me:

Michael "Fritz" Fritzius

Next, adjust the script to go to linkedin.com instead and rerun it.

If you've been to LinkedIn before, you might be surprised that it doesn't know who you are. that's expected.

Since watir is bringing up its own instance of the browser, it does so without any saved information. It's similar to opening a window in incognito mode.

Action items: Connect with me on LinkedIn. Then adjust your script to go to www.linkedin.com

Clicking Your First Button

INSPECT ELEMENT

Right click the button itself and click "Inspect Element" in the menu that pops up

IDENTIFY THE BUTTON

The window that shows up is called the DOM, or Document Object Model. The highlighed item is the button you're inspecting.

BUILD A LOCATOR

Next, construct a "locator" for the button. There are examples at watir.com/guides/form-example/

CONSTRUCT THE CODE

Build the logic into your code to actually find the button, using the locator from the previous step.

CLICK THE BUTTON!

Finally: click the button! Put .click at the end of the line of code after you've built the locator.

ENTERING YOUR LOGIN INFO

AFTER CLICKING SIGN IN

On the next screen you'll be prompted to enter your username and password

ENTERING TEXT INTO A FIELD

These elements are text fields, not buttons, so they'll behave a little differently.

Plus how we interact with them will be different.

LOCATING A TEXT FIELD

See if you can construct a locator for each field.

Then set your username and password accordingly.

Use the .set method to inject text into a field.

LOG IN TO LINKEDIN

Finally, log in. There's a sign in button here as well.

Can you use the same locator as last time or do you have to make a new one?

Navigating to a Profile

THERE ARE A COUPLE WAYS TO DO THIS:

USE THE SEARCH BAR AT THE TOP

Search for Fritz by using the search bar. It's unlikely that you'll find multiple matches, so that helps. Using text entry and button clicks, you can navigate to the profile from the search results.

GO DIRECTLY TO THE URL

It's possible to go directly to the URL, similar to how you went to the LinkedIn site itself.

HOW MUCH OF A CHALLENGE DO YOU WANT?

One way is definitely more challenging. We'll leave it up to you to pick.

SENDING A MESSAGE

So far we've covered clicking buttons and entering text.
Now it's time to combine the two to actually send a message.

CLICK THE "MESSAGE" BUTTON

Use the same methods you've learned about previously. This button isn't much different.

ENTER TEXT INTO THE POPUP WINDOW

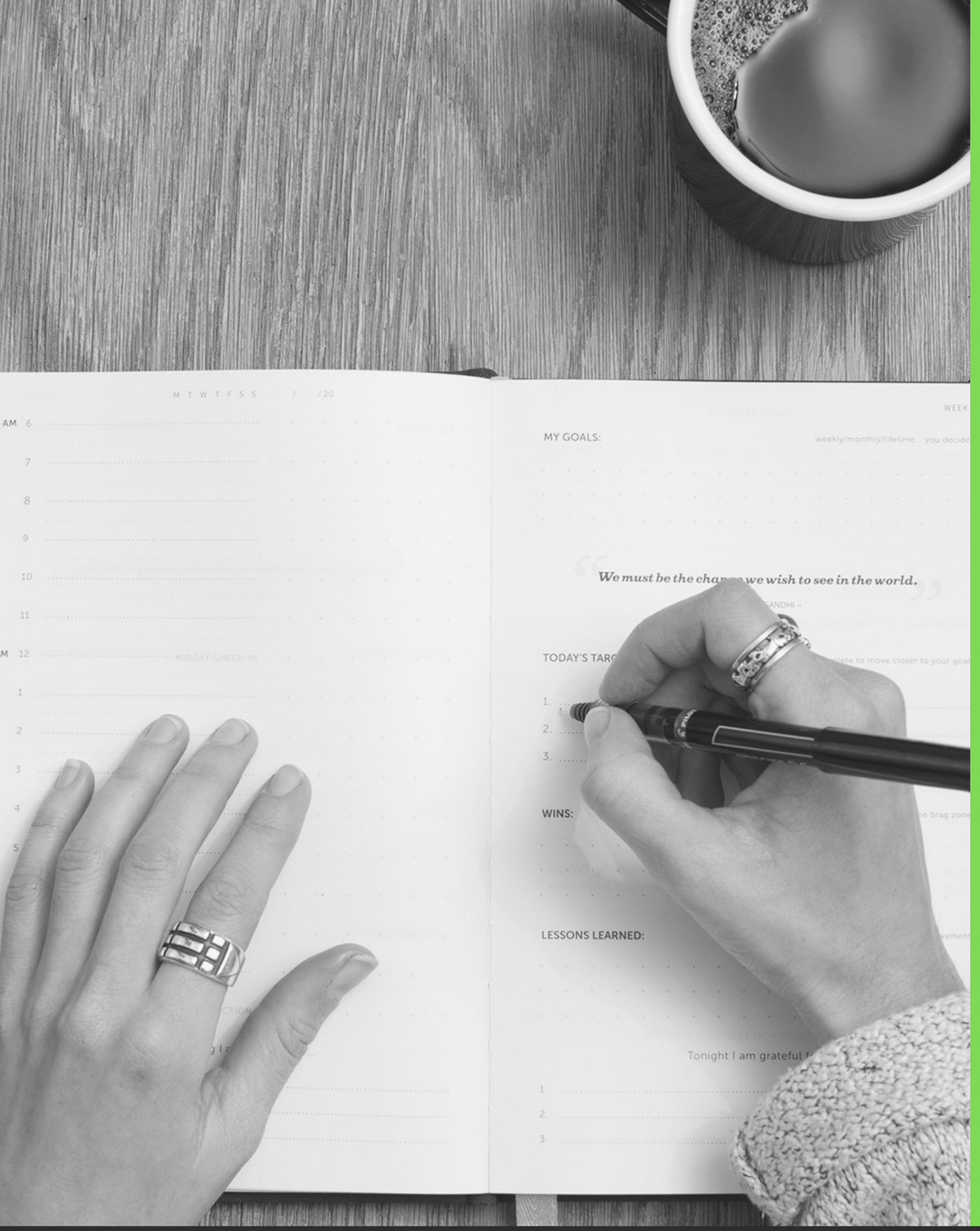
This is another type of text field. It's slightly different though. Hint: you can send key presses using watir. How would you do that?

CLICK THE SEND BUTTON

Land the finishing blow and send the message. Let me know it's from your automated script!

CONGRATS!

you did it!



People often ask,

AND NOW YOU KNOW THE ANSWER!

Whenever people ask how to get started in automation, we always recommend starting with an actual thing that does work.

A small project, like sending a message on LinkedIn is the perfect size and challenge for people just starting out.

This is the first step on your journey. The things you learned here apply right away to doing test automation, robotic process automation, business process automation and all kinds of other ways to save time and then money for a company lucky enough to have a person with your skills.

HTTPS://CALENDLY.COM/ARCHDEVOPS/WORKING-SESSION

Next Steps

WHERE DO WE GO FROM HERE?

ROBUST

Are there places where the automation is a little rickety? How can you improve it to make it more robust?

MAINTAINABLE

Eventually you'll have to change your automation to keep it maintained. How can you make this as easy as possible?

EASY-TO-READ

A month from now, will you be able to read what you wrote? Will other people? How can you write it to make sense now and later?

MODULARITY

Are there parts of the automation that look like they can be reused? Refactor your code to be modular,so you can write less code later.

AUTOMATE!

What else is out there that you can automate? Using what you know now, are there efficiencies you can make with your new found skills? Go find 'em!

HTTPS://CALENDLY.COM/ARCHDEVOPS/WORKING-SESSION

We'd love to chat!

We had a lot of fun putting this together and hope you learned a lot. We'd like the chance to get to know you and your background.

CLICK HERE TO SCHEDULE A CHAT

And if you really are stuck on something related to automation, we're happy to help. We have one-hour working sessions available.

CLICK HERE FOR A WORKING SESSION