Guides **Testing WebDriver Testing** Introduction

Introduction



A CURRENTLY IN PRE-ALPHA

Webdriver support for Tauri is still in pre-alpha. Tooling that is dedicated to it, such as tauri-driver, is still in active development and may change as necessary over time. Additionally, only Windows and Linux are currently supported.

WebDriver is a standardized interface to interact with web documents primarily intended for automated testing. Tauri supports the WebDriver interface by leveraging the native platform's WebDriver server underneath a cross-platform wrapper tauri-driver.

System Dependencies

Install the latest tauri-driver or update an existing installation by running:

\$ cargo install tauri-driver

Because we currently utilize the platform's native WebDriver server, there are some requirements for running tauri-driver on supported platforms. Platform support is currently limited to Linux and Windows.

Linux

We use WebKitWebDriver on Linux platforms. Check if this binary exists already (command which WebKitWebDriver) as some distributions bundle it with the regular WebKit package. Other platforms may have a separate package for them, such as webkit2gtk-driver on Debian-based distributions.

Windows

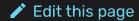
Make sure to grab the version of Microsoft Edge Driver that matches your Windows Edge version that the application is being built and tested on. This should almost always be the latest stable version on up-todate Windows installs. If the two versions do not match, you may experience your WebDriver testing suite hanging while trying to connect.

The download contains a binary called msedgedriver.exe. tauri-driver looks for that binary in the \$PATH so make sure it's either available on the path or use the --native-driver option on tauri-driver. You may want to download this automatically as part of the CI setup process to ensure the Edge, and Edge Driver versions stay in sync on Windows CI machines. A guide on how to do this may be added at a later date.

Example Application

The next section of the guide shows step-by-step how to create a minimal example application that is tested with WebDriver.

If you prefer to see the result of the guide and look over a finished minimal codebase that utilizes it, you can look at https://github.com/chippers/hello_tauri. That example also comes with a CI script to test with GitHub actions, but you may still be interested in the WebDriver CI guide as it explains the concept a bit more.



Last updated on Mar 25, 2023