BrowserStack CASE STUDY

Asia’s Largest Classified Marketplace Uses BrowserStack to Scale Automation

With only a handful of QAs to test through weekly release cycles, Carousell integrated BrowserStack into their pipeline to power up their continuous testing workflows.

THE CHALLENGE

Make release days stress-free.

To meet consumer expectations in the fastest growing mobile market in the world, brands in APAC have to be more than “mobile-first”. They need to keep up with a highly dynamic—and fragmented—mobile landscape.

The largest consumer marketplace for APAC users, Carousell, was releasing new features and updates every week. But release days were stressful for the whole team.

“Everyone dreaded Fridays because it meant unpredictability—about issues we’d find, fixes we’d need to make, and how late it’d be before everyone got home,” says Martin Schneider, Delivery Manager and Senior Test Engineer at Carousell.

To find and fix issues well before release days, Carousell needed foresight into quality throughout their release cycles. They recognized the need for stable automation and continuous testing in their CI/CD pipeline. But getting to this stage was a twofold challenge.

“When we first began automation, it couldn’t really stick,” says Schneider. “With our setup, each team had a dedicated test engineer to test their builds and features.” Carousell changed its ‘dedicated QA’ setup so a group of testers could work exclusively on a stable, high-coverage test automation framework.

However, given the rapid pace of engineering, test engineers were kept too busy keeping up through manual testing. Infrastructure upkeep took up any time that was left.

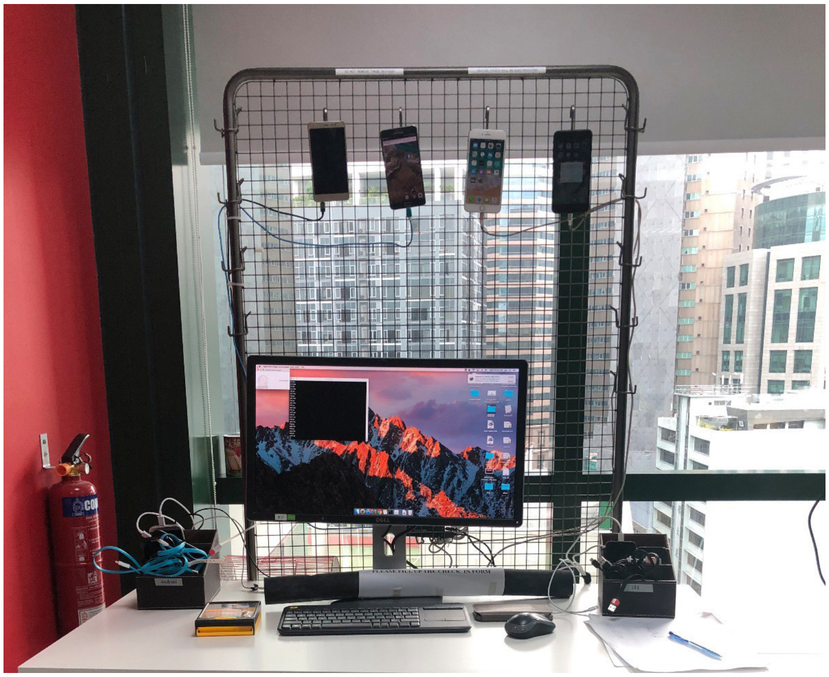
“We ran regression tests once a day on a device lab that was difficult to maintain,” says Schneider. “Making the decision about adding new devices, updating the existing ones, removing the ones that weren’t needed—the small issues became cumbersome once they added up,” he says.

NUMBER OF EMPLOYEES
201 - 500

INDUSTRY
eCommerce

BROWSERSTACK PRODUCTS USED
App Automate

ROI
Reduced manual regression testing time by 7x



Carousell's in-house Device Lab

To give their testers time to focus on scaling the automation framework, Carousell tried to outsource the infrastructure to a device farm offered by their existing cloud services provider.

But, "There were restrictions with this solution," says Schneider. "To test a build, we had to re-upload the app and the tests. For a single test run, the setup and teardown of the test device would take more than five minutes. It added to our turnaround time," he says.

Carousell needed quick, deterministic test feedback throughout the release cycle. To get there, they needed to scale their automation framework—and find a stable, high-performing test infrastructure to run it on.

THE SOLUTION

An out-of-the-box cloud testing infrastructure that could scale on demand.

Due to the restrictions of their current device farm, Carousell began looking for a cloud infrastructure more suitable to automated app testing workflows.

During evaluation, Carousell was leaning towards BrowserStack. "It was one of the more mature solutions," says Schneider, reflecting on the PoC.

BrowserStack's complete suite of test automation products was an added bonus. "Our priority was a stable infrastructure to run app automation, but we also had a vision of what we could achieve with automation

> *"Before BrowserStack, it took eight test engineers a whole day to test.*
> *Now it takes an hour.*
> *We can release daily if we wanted to."*

Martin Schneider,
Delivery Manager,
Carousell

testing on the whole," he says. "We wanted a solution that offered web automation as well—so we could scale to testing beyond Android and iOS to more browser/OS combinations."

Post-evaluation, BrowserStack turned out to be the best fit, outperforming competitors on three key criteria: stability, availability of devices and ease of use.

"Tests should fail only when there's a problem in the app, and not in the automation framework or test infrastructure. If the tests kept failing because the devices are unavailable or not properly reset, we'd be wasting our time investigating the source of a false positive," says Schneider.

"We needed something reliable and quick to get started with. We had to run tests on a large number of devices at every pull request, in a very short time. "

THE IMPACT

Continuous testing through the week and stress-free Fridays.

With BrowserStack, Carousell's test engineers were able to scale their automation test runs by 13x and reduce their regression testing time from a day to an hour.

"From what I see and hear from the team, everyone is a lot happier now," says Schneider. "Before BrowserStack, we didn't have a sense of the quality of release until we actually got to test the release candidate on Friday.

It would take a team of eight engineers almost the whole day to test, find issues, and report results. Then we would make hotfixes and retest. It was a frustrating process of back and forth, and by the time we were sure of the quality, it'd be long past closing hours," says Schneider. "Now, it takes us an hour to run our test suite. We can release daily if we wanted to."

The effect goes beyond release days. Schneider and his team can now inform changes throughout release cycles with reliable test results.

Average manual regression testing time per release went from ~35 hours to ~5 hours.

"The rest is automated now," says Schneider. "With stable automation, we can prevent errors from getting into the master branch and individual builds. Release days are less stressful because testing is balanced throughout the week."

The team currently uses BrowserStack Real Device Cloud to run their regression test suites after verifying pull requests on an internal simulator farm. As the team refines processes, Schneider looks to more ways of scaling with BrowserStack.

Building the iOS simulator farm

“We are evaluating whether it’d be better to run tests on pull requests on real mobile devices instead of emulators or simulators,” says Schneider.

“Eventually, we want to create a culture where quality is everyone’s responsibility,” says Schneider. “We want to get to a stage where test engineers work only on the automation framework: creating new tests, refactoring old ones, tooling, etc.—and everyone can run the tests on their own on a stable, reliable infrastructure.”

Find out how **BrowserStack** can help your dev team test faster!

Visit Us: www.browserstack.com