

A MICROMOBILITY COMPANY WENT FROM 100% MANUAL TESTING TO 100% AUTOMATED WITH AN 80% DECREASE IN BUGS.

Customer Overview

Tembici is Latin America's leading micromobility tech company creating solutions to inspire a revolution in urban space. They have 5 different apps with both android and iOS operating systems. Manual testing took too long and they were looking for a consistent platform for testing native apps. Everyone had their own strategy for testing and they were breaking each other's tests.

They needed a QA system with consistency where testing would not corrupt the QA environment and the same tests could also be executed in production without changes. Tembici needed a tool that was easy to scale, maintain and the engineers could simply use it.

The organization did not want to learn another language or a complex framework. They were accustomed to testing websites and were looking for a system that would also test native apps.

**80%
Decrease In
Bugs**

testRigor enabled Tembici to catch more bugs before release.

**100%
Automated
Coverage**

Tembici was able to achieve 100% automated test coverage.

**Testing 6
Builds in 3
Hours**

Previously, Tembici needed a week to test these 6 builds.

The Problem

Before implementing testRigor the development team was manually testing and the mobile team was the last mile of testing. Every engineer had their own processes for local testing and they were stressed. During each sprint engineers had to stop developing new features so they could complete manual testing. The huge commitment of manual testing included downloading, installing and setting up the apps for each of the tests. This resulted in a lack of engineer productivity since they were not focused on their core role.

Initially they were completing 1 build a week with their manual process and they needed a better solution. With manual testing they were running into more bugs in production and postponed deliveries. They needed consistent quality environments and use cases where they could execute all their tests at once rather than scattered among multiple developers. Prior to testRigor each developer had one device and they had no concept of a device farm where they could test multiple devices and configurations.

They needed to remove the developers from manual testing, reduce stress while creating a consistent organizational testing strategy.

Key Objectives

- Establish a standardized testing process.
- Refocus engineers on developing new features, instead of testing.
- Reduce the number of bugs in production.

“The easier platform to achieve the consistency of maintaining and running the tests in a speedy way.”

 Carolina V.
IT MANAGEMENT

The Solution

Tembici chose testRigor as their automation test system. testRigor made it easy to code test cases where their engineers could use plain english without having to learn any new technical programming languages. testRigor was easy to use and developers could begin making tests within an hour of starting.

Tembici historically tested 1 build a week but now they are doing 6 builds in 3 hours. They are performing more frequent releases and have 0 manual tests in the pipeline. They now see everything they did manually being automated. Additionally they are using testRigor's device virtualization and testing multiple devices without changing their tests. testRigor allowed them to reach full end-to-end testing for the entire app and it brought people together to talk about testing. Implementing testRigor created less stress for the team and gave engineers back time to be more productive.

“It [testRigor] is the better platform to build your UI tests.”

Carolina V.
IT MANAGEMENT

Solution Highlights

- Developers could begin making test within an hour of starting testRigor.
- Able to test multiple devices without having to write new tests.
- Went from testing 6 builds a week to 6 builds in 3 hours.
- Eliminated all manual testing.

Before

Manually collecting screenshots and details for bug reports.

After

Execution log with test details and graphic artifacts provided for each test run in a link.

The Result

The implementation of testRigor decreased bugs reaching production by 80%. Test executions were now done multiple times in hours instead of once a week. The engineering team gained back valuable developer hours and the stress level on the team was reduced significantly.

Furthermore the entire team from product managers, UI/UX to engineers all had a consistent framework to create, communicate and execute automated tests. This ultimately led to a more productive team utilizing a tool that every member could use.

Summary

Objective 🎯	Result 📊
Decrease Bugs In Production	80% Reduction in bugs.
Standardized Testing Method	testRigor brought the team together, now everyone is talking about QA.
Reduce Time Spent Testing	testRigor saved hours of engineering time.
Increase Testing Speed	Went from testing 1 build a week to 6 builds in 3 hours.

testRigor can also help you

“A very productive platform for UI and customer facing testing use cases.”

Roberval R.
DIRECTOR OF TECHNOLOGY

testRigor's no-code automation platform makes it easy for QA teams to quickly build test automation while spending almost no time maintaining tests. Tests are in plain English and empower any person with or without technical knowledge to be able to quickly build and maintain tests as well as understand test coverage.