



CASE STUDY

Who:

With 60 million orders delivered per month, iFood has quickly become the most popular app for food delivery in Brazil

Situation:

Slow build times made the iFood team consider alternatives to their existing CI/CD service

Solution:

A MacStadium cloud built with bare metal Mac minis and Jenkins, leveraging remote caching for iOS builds

Impact:

A 75% improvement in build times led the iFood team to move their Android builds to MacStadium as well

“After a few weeks of real usage of the environments, we understood that we were right by moving to MacStadium.”



www.ifood.com.br



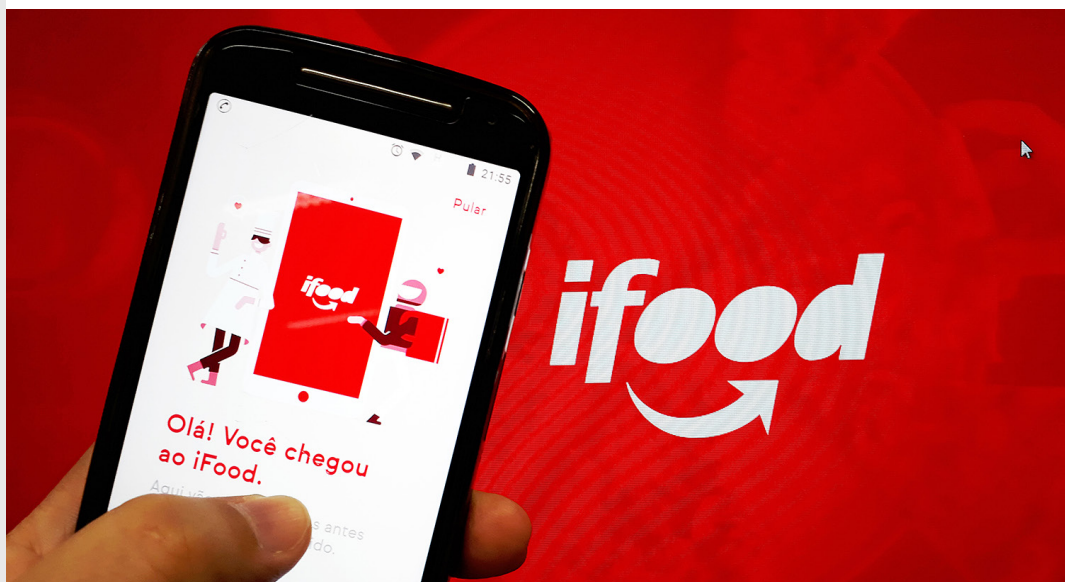
iFood: Better Build Times for iOS and Android

About iFood

The leader in food delivery in Brazil, iFood, has created an innovative system to help customers access takeout and delivery services in their cities. iFood allows users to easily order food from menus that are updated in real-time with accurate prices, descriptions, and exclusive promotions, all without any extra fees.

Before MacStadium

At the beginning of 2019, iFood's mobile DevOps team was using a third-party continuous integration and continuous delivery (CI/CD) solution for both iOS and Android development. Everything seemed to work well, but there was



one major snag - build times. As the iOS and Android pipelines were becoming more and more robust, the team started to see significant build time increase.

The CI/CD team came in to help, but before moving everything from one solution to another, some testing was required. The team planned to migrate their pipelines to Jenkins to allow for greater control. At the end of 2019, iFood began doing tests locally in the office with some spare Mac minis that they had on hand, and after seeing some excellent, real results, the team decided that it was time to scale up. Knowing that they could easily increase their capacity when needed, iFood chose MacStadium for their iOS CI infrastructure.

Moving to MacStadium

iFood started with bare-metal Gen 3 i7 Mac minis for iOS development and EC2 with a c5.2xlarge (8 vCPU + 16GB RAM) on AWS as their Android environment. Once all the infrastructure was in place, the team began to migrate from their previous CI environment to Jenkins. The move would allow a direct comparison between iOS and Android pipelines, even though two separate services were running them. It couldn't have come at a better time since iFood's development team was expanding, and so was their codebase. Now they had a scalable solution and room to grow.

"After a few weeks of real usage of the environments, we understood that we were right by moving to MacStadium," said Diogo Ribeiro Mendes, manager of QA and DevOps at iFood. "However, we didn't have the same success on AWS. The Android builds were having mixed times and were not as constant as iOS."

With iOS builds running well, it was time for iFood to fix their Android builds. They devised a two-pronged approach: first, they created a build.gradle configuration dedicated to the CI, and then

“We didn't have the same success on AWS. The Android builds were having mixed times and were not as constant as iOS.”

Diogo Ribeiro Mendes
Mobile QA & Mobile CI/CD Manager
iFood

they changed their AWS instance to a c5.4xlarge (16 vCPU + 32GB RAM), doubling what they had initially.

In order to keep everything up and running, the team continued to run parallel builds for iOS and Android. Despite seeing some improvements with Android during those weeks, the team started to experience some strange daemon behaviors caused by the parallelism.

The two-pronged approach was quickly scrapped for a new, unconventional plan: move all the Android builds to Mac minis at MacStadium. This meant doubling their current MacStadium infrastructure.

"We switched our Mac mini G3 at MacStadium to the newest ones available, the Gen 4 i7 six-core with 32GB, and integrated our Android builds to those bare metal minis as well," Mendes said. "And everything became stable on Android as well - thanks again to MacStadium."

45% on iOS and 35% on Android was enough build time improvement needed to feel confident about migrating

The team continued running daily pull requests and release tests in the iOS and Android environments on the minis at MacStadium. iFood noticed a tremendous build time improvement: 45% on iOS and 35% on Android. That was all they needed to feel confident about fully migrating to the new environment.

Even More Speed with Remote Caching

Since they had seen such improvement with moving to a bare-metal mini cloud at MacStadium, the team decided to leverage

“ We integrated our Android builds to bare metal Mac minis, and everything became stable on Android as well - thanks again to MacStadium. ”

that and start using remote caching for both iOS and Android. iFood was able to use the newly improved Buck Cache for its iOS builds and Gradle Remote Cache for Android.

This gave the iFood teams another 50% improvement on build time in addition to the previous 40% gain they got from migrating to MacStadium.

“This is just an example, but we were able to bring the pull requests on iOS from 45 minutes to 10 minutes on the full pipeline,” said Mendes. “Which is a total of 75% better build time, with the improvements we were able to see with MacStadium and remote caching.”

Conclusion

Deciding to move both iOS and Android pipelines to a Mac mini cloud may have seemed like an unconventional plan, but iFood is not the only MacStadium customer using a bare metal mini cloud to build Android apps. With unmatched speed and efficiency, MacStadium was the natural choice for an organization that continues to scale pipelines and development teams. The team’s commitment to improving build times, the developer experience, and enhance the CI pipelines at iFood led to a massive reduction in build times and an increase in stability in both iOS and Android. And because of the move to MacStadium, iFood was ready for the big boom they saw in users during the COVID-19 pandemic and can focus on scaling up even more.

“ [We experienced] a total of 75% better build time with the improvements we were able to see with MacStadium and remote caching. ”

Learn how you can get similar results. **Get in touch** with our sales engineers to schedule a free consultation and sign-up for a free trial of a MacStadium private Mac cloud.
