

# galileo

## Building the Future of Healthcare at Galileo

Galileo is a health technology company that provides affordable, high-quality patient care in a mobile-first model. Their two products, a digital platform for doctor visits, and a value-based care delivery model, are at the epicenter of innovation in healthcare. Patients are able to book appointments, be seen virtually by their doctor, and receive the care they need all through Galileo's interface.

### **Making feature releases developer-friendly**

When asked about Split, Stephen Snow, Backend Engineer at Galileo, mentioned how developer-friendly it is. As Galileo expanded, they were able to manage their growth through feature flags. They make code changes and then coordinate with their marketing teammates when it's time to turn the features on. This makes it easier for the development team because everyone on the team knows what's going out and when. Developers don't have to worry about doing a rollback. The ownership of the feature is then given to a product manager to be able to decide when a feature release will happen. If something goes wrong, all they have to do is utilize the kill switch, turning the feature flag off.

**“Split puts us one step closer to continuous deployment, which is our ultimate goal.”**

Sergio Gutierrez, Backend Team Lead

## Releasing features when users are ready

The code in production without immediately releasing it to users. By separating code deploys from feature releases, they were one step closer to their goal of continuous deployment. The problem they were trying to solve with feature flags was that the developers wanted to deploy as much as possible, but the doctors and healthcare providers hated seeing a lot of change in their system. Using feature flags made everyone happy. The development team was able to continue rapidly deploying code, keeping the feature off for users, and then turning the feature on when it was time.

**“Without Split it was hard to time the release of our web app, API, and mobile app - given app store approvals. Now we can group a change behind a feature flag, do a production dry run, and toggle it on for users.”**

Stephen Snow, Backend Engineer

## Increasing confidence with canary releases

Galileo uses percentage rollouts, or canary releases, for their larger infrastructure project migrations. One of these migrations included transitioning to a new authentication platform. They knew they wanted to upgrade to this new framework, but no one wanted to touch it because of the fear of something going wrong. Galileo notes that their biggest game changer was when they started migrating users to the new system slowly using a canary release behind a feature flag. It gave them high confidence for such a critical capability, whereas before, there was fear around taking on the project. They released this change first to internal users, then tested for volume and performance. As their confidence slowly increased, they were able to confidently allocate the full 100% traffic to the new authentication platform and successfully validate that it performed better than the previous architecture.

**“Split gives us an easy and fast way to pull the plug if something goes wrong without a heavy involvement from engineers.”**

Sergio Gutierrez, Backend Team Lead

## Developing rapidly to meet changing patient needs

COVID-19 has accelerated telemedicine as patients avoid in-person visits to doctor offices, making apps the de facto way to provide and receive care. When the coronavirus hit, Galileo made their product free for everyone and began to roll out services to new states. They implemented these changes through feature flagging. Once the marketing campaign was set up and all the requirements were set for a new geography to expand into, they toggled the feature flag to be on. This caused a huge influx of traffic that they had to prepare for. The development team added the ability to toggle the creation of new medical cases to manage capacity if they found that the clinical team was overwhelmed.

**“With dynamic configurations in Split, we spent less engineering time to open registration to new states, because no code changes were needed to support the influx of new patients.”**

Sergio Gutierrez, Backend Team Lead

Another clever thing they did was they used feature flags to set an alert if there weren't enough doctors on the platform to support the new influx of patients, because they did not increase the clinical team. When they ended this campaign, the only thing they had to do was turn the feature flag off. Due to the complexity of the requirements, they leveraged dynamic configurations in Split, and were therefore able to control the exact experience of their user base. Dynamic configurations reduced the burden on engineering time because no code changes were needed at release time, all the changes were encoded in the feature flag.

Galileo's lean engineering team has risen to the challenge of the shift to telemedicine and a rapid expansion across the United States. With Split, they've rapidly developed new features, while keeping risk and stress to a minimum.