Breno Aquino
Software Developer

GitHub
BrenoAquino

LinkedIn
BrenoAquino

Email
brenoaquino7phr@gmail.com

Skills

Swift | iOS | SwiftUI | Combine | View Code | SPM | Cocoapods | SDK | Git | CI/CD | Fastlane | Github Actions | Bitrise | Python | AWS

About

I am a person who is always looking for innovation and automating everything possible, enabling us to focus on more important tasks. I have experience with projects in the areas of education, streaming and finance. I also have experience in creating SDKs, both for internal use and for sale to third parties. I am always open to discuss about architecture and performance.

Professional Experience

Rakuten, United States/Canada • Remote

iOS Developer

October 2022 - current

- Responsible for creating an analytical SDK for internal use by the company with the aim of obtaining higher quality data to facilitate the mapping of product improvements. As the SDK could be called by any component at any time, it was necessary to create thread safe components, and to ensure this I created tests creating multiple threads and trying to access the resource at the same time. Test coverage for the SDK was around 92%;
- The analytics SDK was also created to support different countries and each with its own domain. There is a buffer to avoid excessive requests and each buffer is linked to the country of the user who generated the events;
- Worked on migrating authentication to using a WebView in which it talked to the application through JavaScript messages;
- Improved the privacy flow with the aim of facilitating expansion to other states/countries that follow different privacy laws;
- Responsible for creating components to serve as bridges between RxSwift - that we are removing from the project - and Combine. The objective was that new features and refactorings do not use third-party framworks, but to talk to existing components, the mentioned classes would be used.

iOS | Swift | SwiftUI | Combine | Unit Testing | XCTest | Xcode Cloud | Fastlane | SDK

Itaú, Brazil • Remote

iOS Developer

March 2022 - October 2022

- Responsible for leading and guiding a Junior developer with an emphasis on iOS development. I was responsible for meeting with him every week to answer questions and generally providing a collection of articles on the study topic selected at the time. We also communicate during the week to help with development and help with code reviews;

- As the iOS team was made up of a significant number of developers and dozens of squads, we used several repositories for each module and a main one with the app focused on integrating and coordinating the app's navigation. All module distributions were used cocoapods;
- To support multiple apps in the App Store and avoid duplicate code, I used different build schemas and configuration to define how some components/modules should behave and look;
- Responsible for creating and maintaining multiple modules related to the app's notification/deeplink flow, the app used to have about 28.000.000 active user per month;
- Worked on redesigning the app, where I could apply improvements to the app's architecture and making it more testable and maintainable.

iOS Swift UIKit View Code Unit Testing XCTest Modules Swift Package Manager Cocoapods Jenkins Fastlane

Meta, United States/Canada • Remote

iOS Developer

December 2020 - February 2022 (part time)

- Worked as a leader creating and defining the base structure project for the client Maple Leaf Sports & Entertainment (MLSE). Defining the priority of functionalities, architectures, APIs and web socket for real-time behavior;
- The aim of the project was to be integrable with any team app. For this, an interface was defined that the host app would implement to configure the SDK theme, being able to modify colors and logo in some locations;
- To respond to events happening in the game in real time, it was necessary to create a web socket connection with the server. With this, the app could respond in real time with: trivia, advertising/celebration banners and mini games - using web view to load them.
- A chat was also developed to send/receive messages among all users, as well as the possibility of creating groups - which would exist only for that game - to send/receive messages;
- Used the RTSP protocol to load Live content, but we also provide compatibility for VOD content. The idea was publish the best moments of the game as VODs;
- As it was an SDK used internally but could be sold for integration into third-party teams, it was necessary to configure CI/CD to publish an already compiled version of the SDK and its documentation in a public repository, along with a sample app to guide integration with the SDK;
- Implement the SDK following MVVM to fulfill the mentioned requirements and provide good maintainability and scalability.

iOS Swift UIKit View Code Unit Testing Streaming Quick Nimble XCTest WebSocket SDK Cocoapods CI/CD Github Actions

Claro, São Paulo - SP, Brazil

iOS Developer

March 2019 - March 2022

- Worked on 2 streaming projects, one created from scratch and another that was already in production - with around 300.000 active users per month. Both of them provided live and VOD contents;
- Implemented Fairplay DRM for live and VOD content. It was necessary to use two different logics since each one worked in a different way. I created an abstraction layer so that the player accepted different DRM implementations;
- Handled different types of VODs such as: multi audio where the audio and subtitle track was disconnected from the media and the user could change any audio/subtitle track without reloading the media; single audio where the audio and subtitles were inserted directly into the media, to change the audio or subtitles it was necessary to load a new media with a pre-defined configuration, it was not possible to choose just the audio or just the subtitles, the user was forced to change both tracks; DVB where the subtitle was received in base64 and it was necessary to display it as an image over the content;

- Implemented an architecture with multiple repositories to facilitate maintainability and scalability. There was a repository containing all the business rules and this was imported - using swift package manage - both into the iOS app repository and would be imported into the Apple TV app when it was developed;
- All new features were covered by unit tests and aiming for 80% coverage;
- Implemented a feature to list all channels and their programs, to improve the performance and the time to render the screen to the user, I implemented a custom cache algorithm to avoid unnecessary recalculations and binary search to improve the time to find the current program based on the time.

iOS Swift UIKit View Code Unit Testing Streaming Quick Nimble XCTest Swift Package Manager CI/CD Fastlane Bitrise

Education

University of Fortaleza - UNIFOR

Graduation, Computing Engineering

2015 - 2019

- Final Project: Research using recurrent neural networks to predict cryptocurrency trends.
- Object Orientation class assistant in 2016.
- Microprocessor class assistant in 2017.
- Researcher in the area of sensor network security, trying to implement an algorithm to detect attacks.