

BRINGING TOUCHLESS PAYMENTS TO 5000 GAS STATIONS



PROJECT

Brazil's largest gas retailer wanted to gain a competitive edge by offering a revolutionary way to pay at its 5000+ gas stations. The solution needed to help mitigate revenue loss as a result of gas purchases with stolen credit cards. The company turned to Clearbridge to build a first-of-its-kind "touchless" payments system and in-store m-commerce shopping experience.

SOLUTION

Clearbridge, in partnership with PayPal, created a full-stack solution (mobile apps and web services) that allows customers:



- » To pay for gas without ever taking out their wallet or mobile phones
- » To pay for other items in-store via barcode scanning and in-store kiosk systems

We delivered the MVP in 6 weeks, with nationwide rollout scheduled over the following 3 months.

KEY TECHNOLOGIES & PLATFORMS

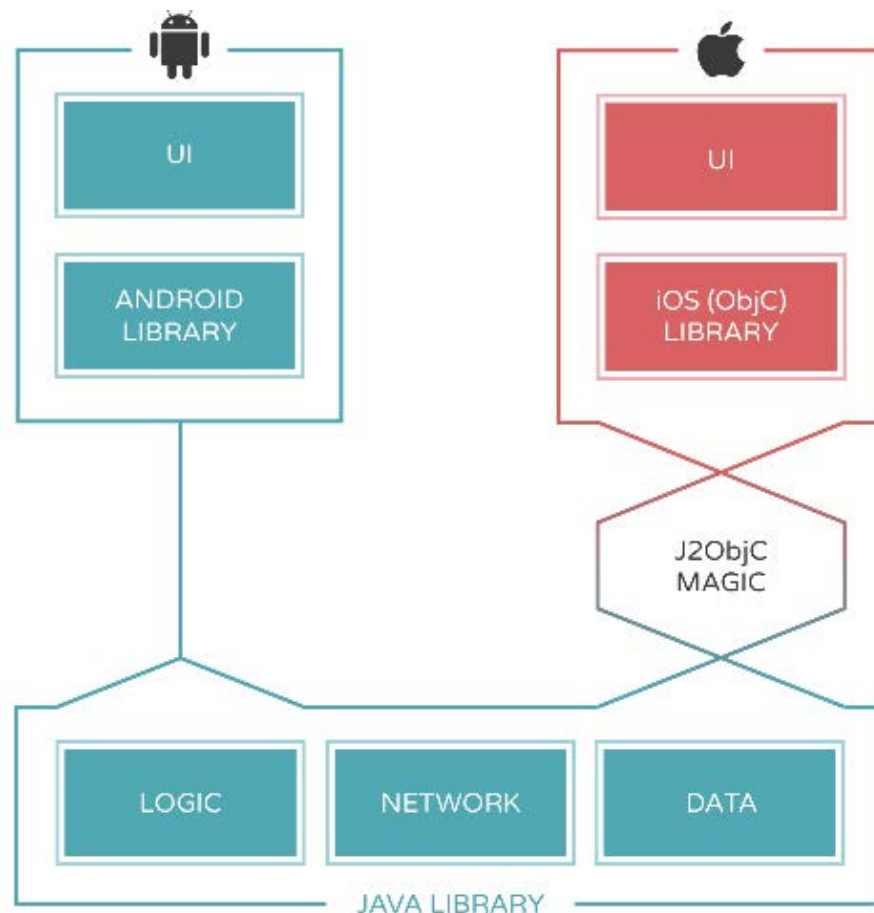
- » PayPal SDK
 - Highly secure and mobile-friendly solution
 - Fast, reliable, scalable
- » GPS Triangulation & ALPR
 - Pinpoint user location at specific gas pump
- » J2ObjC
 - Reduced development time by 25%
 - Reduced QA time by 50%
- » Heroku/Node.js/MySQL Custom Built APIs
 - Custom built web services that communicated to the company's Maestro services, PayPal's authentication and payment services, and the mobile apps
- » ZXing
 - Library used to integrate barcode scanning into the Android app

TECHNICAL & IMPLEMENTATION CHALLENGES

1) TIME TO DELIVERY

Clearbridge had a total of 6 weeks to develop native applications for both iOS and Android. The complexity of the project and the stringent deadline required that Clearbridge find the most efficient development method that would not sacrifice product quality.

We solved this issue by using J2ObjC, which allowed us to code all of the business logic only once, as opposed to separately for each platform.



2) NAMING CONVENTIONS

With J2ObjC, the UI still needs to be coded natively for each platform, meaning UI developers need to be able to read and understand the code. To avoid naming collisions when the code is converted to ObjC from Java, the J2ObjC conversion engine provides very descriptive names for classes, methods, variables, etc. However, this reduces the readability of the code – it will make sense in Java, but not ObjC, which provides very generic names.

Clearbridge was able to customize the naming conventions to make ObjC code cleaner and more readable, making troubleshooting and building platform specific functionalities easier.

3) GEO-LOCATION

The application needed to be able to communicate a user's location to the specific gas station and the specific gas pump being used. In order to complete the touchless transaction, the server needed to know the user's location.

Clearbridge implemented a solution that detects the user location and confirms with the gas station that the user is present via custom built APIs. In case geo-location was unavailable, we created a code generation service which allowed users to input a time-sensitive, temporary code into the app in order to authenticate the payment.

4) LOCALIZATION

The solution needed to support multiple languages on multiple platforms to accommodate the needs of the end users.

Clearbridge built a Node.js script that generated platform specific localization files within seconds to serve the right content to the right user.



CONTACT

REPRESENTATIVE

Sean Huynh

Account Executive

sean@clearbridgemobile.com

647 • 361 • 8401 X 179

WEBSITE

www.clearbridgemobile.com