



WE POWERED VERIZON'S MOBILE SHARE SYSTEM



Objective

Create a mobile environment for users to locate and share transportation in cities.

Solution

An integrated mobile platform prototype utilizing Verizon's Share platform and geospatial technologies that exhibited the desired results.

LEVERAGED TECHNOLOGIES

- Amazon Web Services
- Docker
- Node.js
- React.js
- MapQuest API

PLATFORM INTEGRATION

- Proprietary Share Core System
- IoT integration with OBD-II in Car Device
- Checkfront Reservation System
- MemberMouse - Member

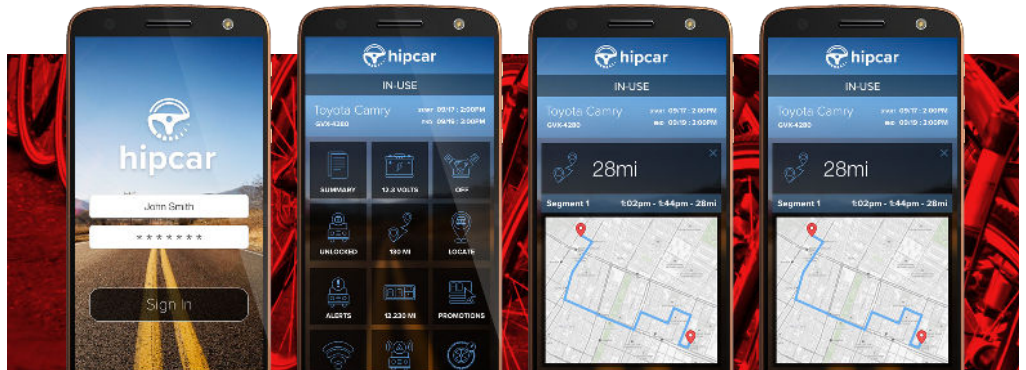
CORE CAPABILITIES

- Mobile Application Development
- IoT
- Geospatial
- Cloud (AWS)
- UI / UX



The Seisan team delivered above and beyond the agreed upon solutions. The dedication, commitment and completeness of the solution and services through Seisan is outstanding."

Shameem Akhter, VP of Business Development, Verizon



Seisan combined our expertise with geospatial technology and IoT devices to create a sharing platform for Verizon to offer its clients.

These days, consumers have a variety of mobile apps to choose from when searching for a car to rent. But how about one that will not only locate a vehicle, but unlock it and manage the rental experience, too? That's the technology solution Verizon was after when they engaged Seisan to develop a versatile platform capable of facilitating back-end reservations while featuring a sharing management system to allow a mobile application to access asset information.

It's part of a plan to enable Verizon clients to gain new revenue by making their assets –

specifically, fleet vehicles that are not in use – available to be rented by the general public. An end user simply needs to walk up to the vehicle that they have reserved and, once they are in close enough proximity, gain access by unlocking it from the mobile app.

Moreover, the app serves as an information hub during the rental period, providing the consumer real-time data such as miles driven, trip log and current statuses such as battery and fuel levels.

In the end, Seisan was able to deliver a solution that allowed seamless data to flow between a variety of smart devices and Verizon's back-end sharing platform.

SEISAN'S APPROACH / VERIZON'S HARMONY SHARE APP

PROJECT OVERVIEW

Verizon engaged Seisan to define, design, develop and deploy a new asset sharing platform solution, called Project Harmony. Seisan partnered with the Verizon product team responsible for the sharing platform to evaluate 3rd party vendors. Our team defined the integration points between those platforms and the existing Verizon vendor technology that provides vehicle IoT data. The objective platform would provide Verizon's small to mid-size fleet customers a method to generate new revenue from vehicles not in use to other end consumers. By leveraging available depreciating assets during periods of inactivity by the main clients, those companies can recoup some costs of ownership. This concept would emulate Zipcar or Airbnb as temporary consumer lending platforms.

TECHNOLOGY DETAILS

After evaluating 3rd party products, Seisan established the development project plan. Using a microservice design built with loosely coupled Docker containers creating flexible options for maintainability and scalability.

By selecting MongoDB with a node interface, the containers instantiated could easily change out the persistence layer if desired, i.e. moving to a SQL database or moving to a different language, such as Java, C# implementations. Creating a flexible design allows the end customer to change out specific parts without having to rewrite the entire application, resulting in quicker development and faster rollouts of new features, functions, and design alterations. Inside the containers we utilized a Node.js Express app for easy load balancing, performance, and maintainability. Node.js Express also allows for a smooth change out the transportation layer, giving Verizon customers the ability to switch to their preferred service e.g. Azure Service Bus or AWS Kinesis.

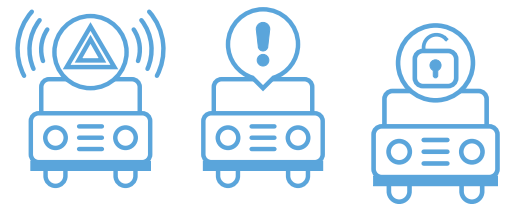
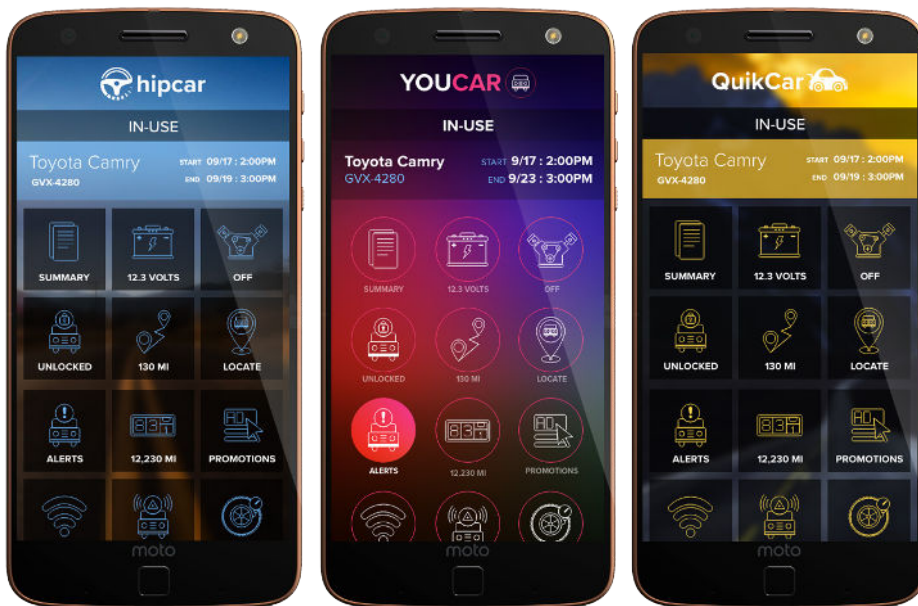
Seisan architected a new platform service layer to tie together a 3rd party reservation solution and separate membership management system that integrated with the vehicle's IoT platform. Seisan also developed a web services layer to support end-user mobile applications. These new services allow end-users to access rental vehicle data in realtime data from their mobile device, such as battery level, fuel

level, current odometer reading, engine on/off status and additional statics on vehicle usage. Other functions provide the ability to locate the vehicle and review trip logs during the rental period.

Seisan developed the core iOS and Android applications that are designed to be white labeled for Verizon clients. The mobile applications were designed to be customized by changing which features and what order they show up on the home screen grid for each client, support the sensor data reading functions and actions available on the IoT device.

RESULTS

The final delivered product was a platform that could be deployed via SaaS model, customer cloud or on-premise solution. Ultimately, Seisan's efforts on Project Harmony forged through the idea of leveraging commercial fleet vehicles on-board tracking and other IoT sensor devices as a way to recoup depreciating asset cost into a reality. Our end product provided additional revenue generating opportunities for our client, Verizon, and by extension their customers.



Lottie Icon Animations

Verizon requested editable animated icons to help their application stand out from competitors. To allow for easy customization, Seisan used Lottie, an open-source animation tool. Lottie uses animation data exported as JSON files, allowing users to update the look of their app by simply changing the Hexadecimal color values in the code.

White Label Branding Made Easy

The Verizon Harmony Mobile Application was created using editable styles and tile-based icon approach. This allows companies to easily rebrand the app to match their specific visual style and business offering.

