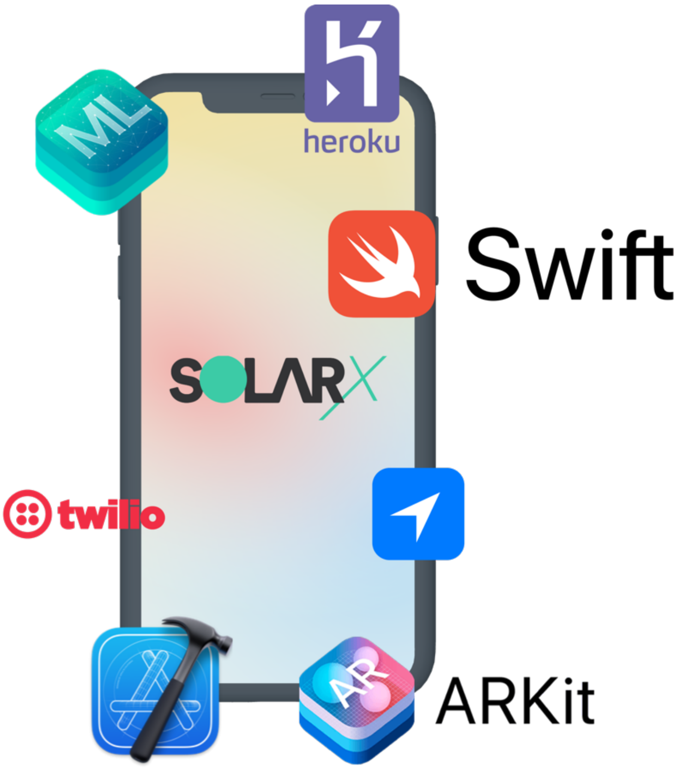
SOLARX

by The Misfits

Technical Overview

Our application, *SolarX* is served as mobile-first application. Primarily developed using *Swift* in *Xcode*, the application is built around principles of user-centric design while solving the problems regarding accessibility and availability of clean and sustainable energy.

Using Machine Learning and location-based inputs, the app adapts to the user, providing them accurate suggestions regarding eligible subsidies, contractors and products that are tailored specifically to match their requirements.



Libraries, toolkits and frameworks used

AR, Map - Cost Estimation

The estimation models are built around the user providing accurate roof measurements, geolocation and economic parameters, for the app to personalize Solar suggestions for them.

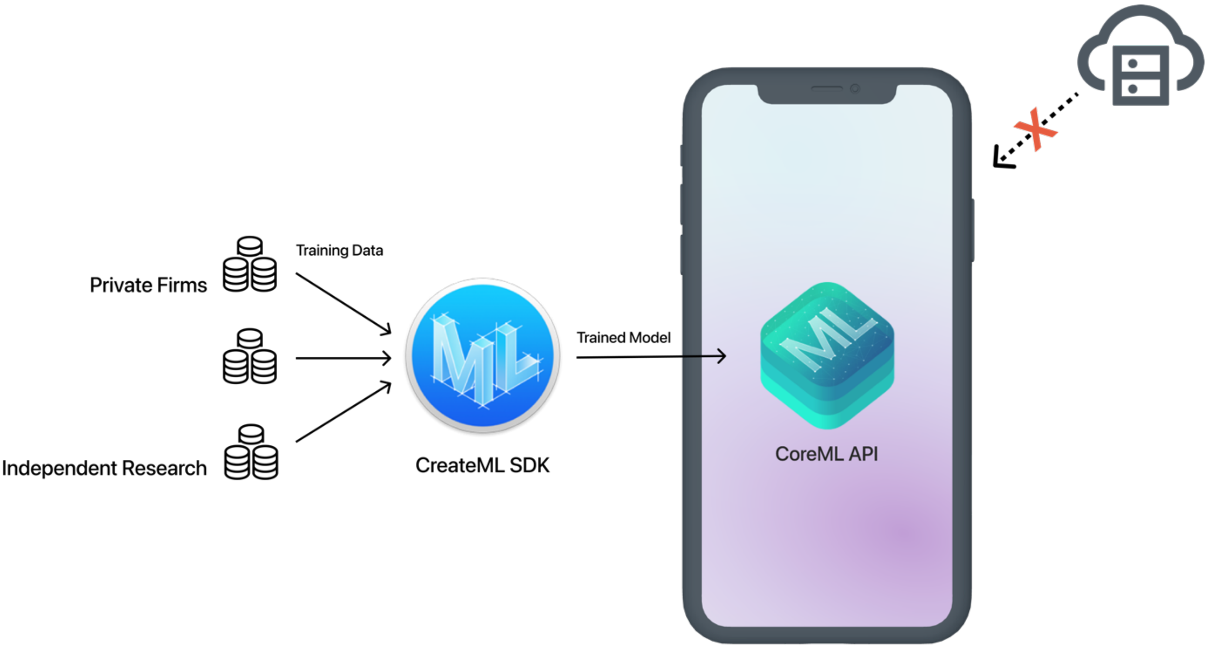
To aid in roof measurement, three options are at offer – Augmented Reality based scanning, area selection on satellite map view, and Manual entry of dimensions. AR and Map provide an interface for highly accurate, scalable and large-scale measurements.

Native APIs – *ARKit* and *CoreLocation* are used for the same, resulting in highly accurate and fast measurements.

Native ML - Offline Capabilities and Privacy

To cope with issues regarding internet connectivity and limited bandwidth, SolarX core features like Installation cost estimation, carbon offset calculator and the ML model concerning monthly savings are all made completely functional without the need of a working internet connection.

To achieve these offline capabilities, we have used native *CoreML* API in the iOS SDK. This not only provides round-the-clock access to our services, but also enhances privacy and security of users by offsetting the need to rely on server-based API communication to a Machine Learning model hosted somewhere in the cloud.

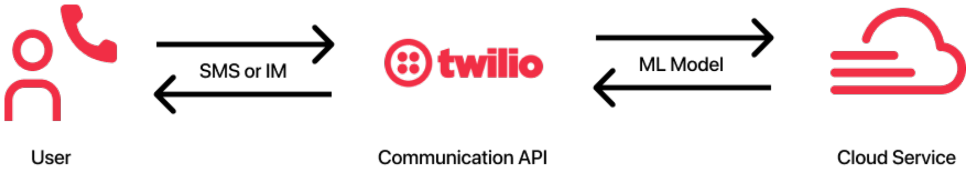


Native and secure Machine Learning pipeline

This secure pipeline keeps sensitive user information like roof images, geolocation and socio-economic demographics on their device, giving them complete control over their privacy.

Communication APIs - Feature Phone Support

For catering to the segment of population that doesn't have access to a smartphone with AR and ML-processing capabilities, a lite version of the application's Installation Cost model will be deployed on a cloud-based communication service like *Twilio*, extending support to SMS or Telegram-based chat bots that rely on textual input to return installation cost estimates to the users.



IM and SMS communication-based cost estimation