Tell us what your idea is.

Describe in 250 words what the feature or service will do and how you'll use Machine Learning to push the bar:

Drivety: A smart assistant app connected with IoT Android Things device to secure inside car activity.

With the TensorFlow Lite models integrated, Drivety unlocks new experience where every car can have the same high-end security features now. Drivety offers:

[1] Seatbelt Worn Detection.
Detect and trigger informative alarm rather than **"Beep"** boring low-level notification sound in some common cars to increase awareness of some ignorant people about the importance of wearing a seat belt.

[2] Crime Detection.
Detect and immediately sends payload notification to the police station and nearby Drivety user for early rescue. Potential victim user can also special trigger this feature by shouting: **"Drivety Help!".**

[3] Smoke, Heatstroke & Dangerous Chemical Gas Leak Detection.
With **BME280** sensor installed on IoT device, Drivety scans heat temperature, air quality, and visible smoke surrounding inside car. Immediately sending notification to car owner and nearby Drivety user for early rescue. This feature hopes can **minimize** even **reduce** incident cases where kids died left by their family inside the car every year into **0**.

[4] Potential Car Theft Detection.
Detect and immediately send payload notification to the car owner confirming the car is being driven by an unrecognized driver. If owner confirms **"Block!"**, Drivety immediately sends payload notification to the police station for early action.

There are still a lot of other features we can integrate inside Drivety:

- **Oil & Fuel Management Prediction Based Distance Record by Google Maps.**
- **Drunk / Fatigue Driver Detection.**

But to keep it simpler, Drivety is satisfied with those 4 major features for now.

Tell us how you plan on bringing it to life.

Describe where your project is, how you could use Google's help in the endeavor, and how you plan on using On-Device ML technology to bring the concept to life. The best submissions have a great idea combined with a concrete path of where you plan on going, which should include:

- *(1) any potential sample code you've already written,*

[It's been included in repository already]

- *(2) a list of the ways you could use Google's help,*

[1]
Mentorship for an efficient way to identify activity inside car based from real-time video input data in certain frame rate with local models / or downloaded from Firebase.

[2]
Collecting a set of categorized crime images for 2nd feature sure is challenging but still acquirable.

Maybe I could get support 😎 from Google but with much better data quality to optimize the model as well.

[3]
So far I could only afford standard IoT components for prototyping:
- **Raspberry Pi NoIR Camera Module V2 - 8MP** for recording (support night time).
- **BME280** sensor for environmental monitoring.

Maybe Google could recommend other high-end similar components to boost Drivety's IoT performance at its finest but the APIs are still in Android Things scope.

[4]
I'm planning to integrate **BME680** / or **MQ-X** sensor to support gas leaked measurement. However, Android Things 1.0.15 supports the one and only contrib **driver-bmx280** for environmental measurement (temperature + pressure + humidity).

Maybe Google could recommend better approach idea to bring this **Dangerous Chemical Gas Leak Detection** idea to fruition.

[5]
I'm truly lack of experience especially when dealing with how to bring prototype into production phase since I'm just doing prototyping based on SOM and Carrier Board shipped with Android Things Starter Kit all this time.

Google's help here would be greatly appreciated in mentoring design of this Drivety's IoT companion hardware into customized, compacted, and also networking to vendor / or manufacturer to have the hardware assembled based on generated hardware design files: **Schematic & Layout.**

● *(3) as well as the timeline on how you plan on bringing it to life by May 1, 2020.*

Here's the toolkit to bring Drivety App and its IoT companion device (prototype at minimum) idea to fruition.

Software:
● IDE Android Studio.
● Kotlin Programming Language.
● Categorized Car's Inside Activity Image Datasets.
● TensorFlow (TF) Lite + Converter.
● Pre-trained TF Lite **MobileNet_v1** model.
● Firebase as TF Lite custom model hoster and payload notification sender.
● SpeechRecognizer API.

Hardware:

- microSD card with Android Things Image.
- Raspberry Pi 3 Model B.
- Raspberry Pi NoIR Camera Module V2 - 8MP (support night time vision).
- BME280 sensor.
- Breadboard.
- Some LEDs as indicator.
- Some Jumper Wires as connector.

The development journey will start from:

[December 2019]
- Collecting the one and only precious image datasets (daytime and night time modes). Some will be modeled on my own [1st, 3rd, and 4th features] and some [2nd feature] from the police station.

  Disclaimer:
  Normalizing the dimension into **960x480 pixels** will have satisfied quality for processing enough.

- Getting the App UI/UX design ready. This will be collaborated with my UX specialist friend.

- Setup above prerequisite toolkit:
  - Retrain MobileNet_v1 model with own set of images and generate .tflite model files,
  - Setup Firebase and upload the custom models,
  - Assembling components on IoT device,
  - Deciding Android architecture pattern (MVVM will be the choice),
  - etc…

- Some code spike blocks for functionality experiment:
  - Callback retrieving image from Raspberry Pi camera module,
  - Callback retrieving environmental measurement result from BME280 sensor,
  - Callback retrieving voice recognition result from SpeechRecognizer API,
  - Callback retrieving prediction results based from downloaded / local models.
  - Callback retrieving Google Maps coordinates.
  - Callback retrieving payload notification sent by Firebase.

- Establishing Bluetooth connection between App and IoT device,
- etc...

[January 2020 - Mid February 2020]
- **Wiring Things Up!**
  Time to wire things up based on written code spikes and finishing the Mobile App interface.

- QA Testing.

[Rest of the Time - 01 May 2020]
- Let's see if there're insights, recommendations based from **"a list of the ways you could use Google's help"** section to optimize Drivety into way better.

- Publish **Drivety** to Google Play Store.

Tell us about you.

A great idea is just one part of the equation; we also want to learn a bit more about you. Share with us some of your other projects so we can get an idea of how we can assist you with your project.

Hi, My name is Erick Sumargo. Developing Android since 2016 and now I'm working as Android Developer at Indonesia's e-commerce unicorn startup: **Bukalapak.**

Professional Career Timeline:
[2016 - 2017]
Startup developing educational app where students can post any question and private teachers will try to help by solving them in detail: **Solvin** (Inactive, Removed).
https://play.google.com/store/apps/details?id=id.solvin.dev

[2018]
Startup developing educational app where students can have real-time chat with private teachers to consult about their homework or exam preparation: **Newt** (Inactive).
https://play.google.com/store/apps/details?id=app.newt.id

[2019]
Google Certified Associate Android Developer.
https://www.credential.net/ng50pnnd

[Present]
Actively tweaking IoT Android Things platform and Machine Learning integration.
https://github.com/ErickSumargo/AndroidThingsFirebase

https://github.com/ErickSumargo/AndroidThingsCamera

https://github.com/ErickSumargo/AndroidThingsMQTT

Next steps.

- Be sure to include this cover letter in your GitHub repository
- Your GitHub repository should be tagged #AndroidDevChallenge
- Don't forget to include other items in your GitHub repository to help us evaluate your submission; you can include prior projects you've worked on, sample code you've already built for this project, or anything else you think could be helpful in evaluating your concept and your ability to build it
- **The final step is to fill out this form to officially submit your proposal.**