# Introduction to Visual Wake Words (VWW) Application

You've now seen the full TinyML flow in the context of a Keyword Spotting application. Noting particular challenges with preprocessing audio data, training TinyML models, and designing effective end-to-end systems and metrics. You then explored the many challenges with (and solutions for) creating an effective TinyML dataset.
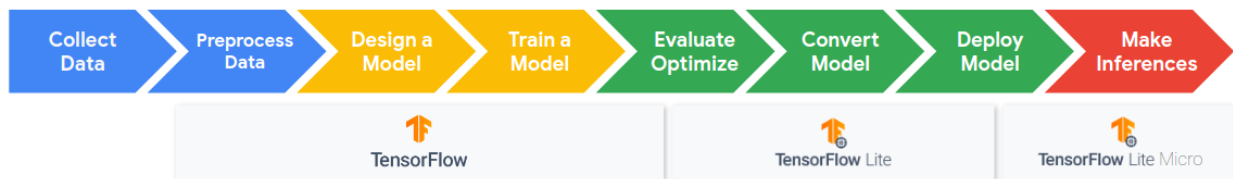
What's the Focus in this Module?

In this section we are going to build on that learning by exploring the TinyML flow and data engineering again in the context of a Visual Wake Words application, focusing on some unique challenges presented by this computer vision application. Visual Wake Words represents a common TinyML visual use case of identifying whether an object (or a person) is present in the image or not.

What's New and Different?

Following what we did for Keyword Spotting (KWS), we will look at several interesting perspectives on the end-to-end VWW TinyML application pipeline. We will understand the characteristics of the camera sensor and how that affects our preprocessing pipeline. We will look into the volume of data generated by the sensor and how that affects our downstream ML workflow. We'll explore a fundamentally different way of doing neural network calculations to make things tiny so they are more efficient on MCUs. Specifically, we will introduce Mobilenets, a new class of models that leverage **Depthwise Separable Convolutions** to use less memory and reduce the total number of computations. We'll then discover a whole new way to train models, **Transfer Learning,** which builds off of an existing pre-trained model to drastically reduce training time. Finally we'll **explore the different end-to-end systems challenges and metrics** present in this novel application. We hope you enjoy this section!