Homework 6. Evaluate a language for a garage-sale buyer app

[131 home > Homework]

Motivation

You're in a team working on GarageGarner, an application intended for people who buy things from garage sales. The old version is web-based, and most of the computation is done in a central server with only the user interface running in users' cell phones. Your boss wants a new version that is less klunky, and is willing for you to develop a native app to do that.

One of the features your boss wants to have is to let the user walk into a garage sale, take a panorama of the items and price tags on display, and quickly report to you on what the best deals are. Your current version does something like this by shipping images to the central server and having it do all the work, but this is too slow and chews up too much network bandwidth. Your boss wants most of the algorithm to run in the phone, to make the application faster. Much of the algorithm's difficulty revolves around the fact that it uses machine-learning based models to process the images, and these can take some CPU time to evaluate. However, newer high-end phones such as the iPhone XS and the Pixel 4 have AI accelerators for algorithms such as yours, and your boss wants your native app to use these accelerators if available.

Your team is investigating alternative technologies to implement the new version of the application. You've selected TensorFlow Lite for running your model on cell phones, but you need to choose a technology for writing and running your app's user interface. You've been given the job of looking into the Flutter user interface toolkit as one of the technology alternatives. Flutter apps are written in Dart, a language that can be compiled to the ARM machine code used in most mobile devices. Your boss suggests that the application use tflite, a Flutter plugin for the TensorFlow Lite API.

For the purpose of this assignment, assume Dart 2.7 and Flutter 1.12 (2019-12-11), along with tflite v1.0.5 (2020-02-26).

Assignment

We are not expecting working prototypes for the proposed application.

Do some research on Dart and its support software as a potential platform. Your research should include an examination of the language and system documentation to help determine whether it would be effective. We want to know whether Dart supports the proposed application well.

Evaluate the features of Dart version 2.7 for the proposed app. Compare and contrast them to the features of Ocaml, Java, and Python that you used to do earlier assignments. What strengths and weaknesses do you see for Dart, compared to these other programming languages? What are the biggest problems you anticipate when using Dart for this application?

Write an executive summary that compares and contrasts Dart to the other languages, and gives Dart's strengths and weaknesses, along with problems for this application. The summary should be in 10-point font or larger and should be at most three pages. You can put references and appendixes in

later pages, if you can't get under the page limit: the appendixes should contain any source code or diagrams. Your summary should focus on the technology's effects on ease of use, flexibility, generality, performance, reliability; thie idea is to explore the most-important language-relevant technical challenges in doing the proposed app. The summary should be suitable for software executives, that is, for readers who have some expertise in software, particularly in managing software developers, but who are not experts in Dart. Please keep the resources for written reports and oral presentations in mind, particularly its rubrics and its advice for citations to sources that you consulted.

Submit

Submit a file `hw6.pdf` containing your summary.

© 2020 Paul Eggert. See copying rules.

$Id: hw6.html,v 1.71 2020/03/05 02:17:32 eggert Exp $