

# Evaluation Various Languages for a Garage-Sale Buyer App

## 1. Abstract

The motivation of the evaluation is to design an application which is intended for people who purchase items from garage sale. The current version of the application is web-based and the related important computation is being accomplished at the central server while only the user interface are being displayed at the user screens. The intended new version of the application allows the users to take a picture of the intended purchasing item and quickly report the price tag of the item and the best deals compared with other products within the sale.

## 2. Introduction

The new version of the implementation should speed up the application faster by running the machine learning based algorithm and models on the user's individual phone. The users phones consist of AI accelerators which then takes part of the CPU time to evaluate the computation and models. Our goal is to investigating different technologies in terms of implementation of the new version of the application. In terms of the investigation, we will be focusing on TensorFlow Lite for running the model on the cell phones. We have also chosen the Flutter which is written in terms of language called Dart as the toolkit of the user interface and a Flutter plugin called tflite for the Tensor Flow Lite API.

## 3. Dart

### 3.1 Examination of the language and system documentation

Dart is one of the general purpose programming language that is origin from Google. The main purpose of the language is to develop application for web, servers and mobile devices. In generally, Dart has a similar structure as C and is capable to transfer to Javascript source code through transcompile tools. The language has supports for different features such as mixins, interfaces, optional typing, abstract classes and reified generics. In order to run Dart, it is requires to compile to Javascript source code in order to run on the Web servers by using the compiler called dart2js. Therefore, the compiled code is able to run faster compared with Javascript due to the fact that it attempt to decrease the number of any exhaustive operations. It is also able to run on the Google Chrome Web Browser.

### 3.2 Support of proposed application

### Mixins

It is a way of reusing code in multiple class hierarchies without multiple inheritance. The purpose of mixins is attempting to divide the general functionality into several small subsections of code which could be reuse in different purposes when we need to use it. When we want to include mixing type, we just need to type the with keyword on the class that we want to applied. The difference between extends and mixin is we can include one or more mixins in a class we wish.

### Interfaces

The main function of the interface is to define the available functions which could be applied on an object. In regards to declare interface in Dart, it does not have a clear and direct syntax of accomplishing that. However, class declaration in Dart are represented as interfaces. Whenever a function defined in the interface, the corresponding class must redefine each of them in order to correctly implement the class.

### Optional typing

Dart is dynamically typed which means it is not necessary to specify type annotations during compile time and successfully run them. Dart has a static checker which is responsible for checking potential problems which are likely to real problems during compile time, but programmer will still be able to successfully compile and run the code.

### Reified generics

During run time, objects created in Dart could carry type arguments under the idea of reified generics. This is considered as one of the runtime operation by providing generic type constructors with type arguments. However, it is optional to provide type parameters when it comes to create instances from generic classes.

### 3.3 Advantages

#### Flexibility

Due to the fact that Dart could be easily run on any application platform such as Android, iOS... Dart considered as a programming language that has high flexibility and portability. Since Dart follows C style syntax, it makes it easier to learn and use the language once the programmers already have the concept of object oriented principles. Compared with Python, Dart also supports asynchronous programming with the functionality of async and await. Dart allows supports unit testing which means programmers are able to select specific sections of the source code to conduct testing and debugging without installing or importing any extra libraries or modules.

Compared with Java, Dart programmers are able to fix the bug and report it immediately without having to

## Evaluation Various Languages for a Garage-Sale Buyer App

handle issues regards to licensing and payment. As a result, Dart is considered as one of the language that work with open source environment.

### Ease of learning

Since Dart supports type inference and it is stoically typed, it makes the programmer task easier when it comes to development by not providing the type annotations. This means that any expression in Dart will be automatically detected in regards to its own data type. Also, the syntax is similar to Java and C. Programmers who are familiar with those languages will learn Dart easily. Dart does not require any specific installation in order to get started on working with the environment. Programmer could just test out the language on the Dartpad platform.

### Robustness

Since Dart syntax is based from C programming language, it compiles really fast due to the fact it is a general-purpose programming language which means it is intended for working with different domains of application. Compared with C, Dart is also a compiled language. However, Dart run faster compared with Java performance. Aside from this, Dart is efficient and reliable because it is type-safe and capable to be compiled with both JIL. JIT (Just in time compilation) refers to performance optimization during run time for any compilation that is Java based. JIT will work with JVM and compile byte code into machine code. As a result, hardware is able to execute machine code directly to gain execution speed.

### Productivity

Compared with Python, Dart also comes along with a sufficient large libraries and packages for programmers when it comes to development. As a result, it definitely speed up programmers work progress by utilizing the embedded functionality in the library. For example, `dart:async`, `dart:collection`, `dart:convert`, `dart:math`, `dart:typed_data`, `dart:io`, `dart:isolate`... For the implementation of the application, the `convert` library could be used to encode and decode different data representation when we need to work with streams. Also, the `io` library will definitely needed for application that are not based on web and this is indeed needed for the garage sale applications. The library could be utilize for working operations related to files, directories, sockets, processes, servers and clients. The `isolate` library could distribute the tasks and request between different threads, but each of the threads communicate through messages instead of sharing memory with each other. The `typed_data` library assists programmers in order to work with data that are fixed sized. The `collection` library provides with different data structures for the programmer to store and process information such as double link queue, hash map, hash set...

### Dart & Flutter

First, Dart is capable of compiling to native code vastly due to AOT (Ahead of Time). Therefore, it speed up Flutter and programmers are free to customize the widgets they are working with on Flutter. Second, Dart is JIT in terms of compilation as mention previously. It particularly speed up the Flutter's stateful hot reload workflow. Programmers could easily modify, reload and execute code through JIT compiler. Third, Dart is able to have smooth and fast transitions for when it comes to create any animations related applications. Fourth, Dart consist of garbage collection and any memory allocation for object without the involvement of locks. It is because Dart uses the idea of generational garbage collection and allocation which means that it is really quick for allocating any objects that only live for a short amount of time. Any applications with Flutter could start up really fast because the implementation is being directly convert into machine code. Finally, Dart does not require Flutter to have different layout language because Dart itself is already declarative in terms of visualization and readability. Therefore, it makes it easier to corporate with Flutter extra and advanced functionality.

### 3.3 Disadvantages

#### iOS platform

The Cupertino widgets are formed based on the iPhone setting. Therefore, it is possible that the design features will not be able to catch up with the latest iOS version that has been released. There is a potential delay problem regards to the application features that are available in Flutter.

#### File Size

The goal of the app developers is to try to keep the implementation as small as possible by avoiding animations, reducing the number of required libraries and packages. However, it is hard to keep the application written in Dart framework small when comparing with applications that are written in Java.

#### Third Party Libraries

Dart is still considered as a relevant new mobile development programming language comparing with C, Python and Java. As a result, it is harder for developers to seek for free outside resources packages that does not come along with the original resource. Therefore, Dart's tool set are still under developing when it comes to seeking for some advanced functionality.

## 4. Ocaml

### 4.1 Advantages

# Evaluation Various Languages for a Garage-Sale Buyer App

## Automatic Memory Management

Programmers do not need to worry about allocating and freeing memory explicitly because the language has a generational and incremental garbage collector that will take care of it. Compared with Java GC, Ocaml GC does not acquire a lot of memory during the start up stage. It uses a strategy of freeing several memory at once in order to avoid the cost of calling the free method multiple times. Ocaml GC has two heaps which are called minor heap and major heap. The minor heap is responsible for storing objects that are small, allocate and freed frequently while the major heap is responsible for storing objects that are long lived. As a result, Ocaml GC is capable moving memory area in order to create a compact heap for making allocation more efficient and faster.

## Efficient compiled code & Portability

Ocaml also support direct byte code compiler and separate compilation. This improves the performances by providing executables that are small by the compilers. It is fast during the compilation process. Ocaml is also capable of running on different platforms, such as Unix, Linux and Mac OS X. Ocaml applications could be compiled into iOS and available into the Apple app store.

## 4.2 Disadvantages

### Lack of Parallelism Support

Ocaml does not have parallelism when it comes to runtime because there is a global lock that stop multiple threads to run at the same time in Ocaml. The lack of parallel computing will cause computation becomes longer without full utilize of the available resources. Also, it might be harder and impractical to solve complex problem which requires more memory and multiple processors.

### Lack of multicore support

Ocaml does not have multicore support which refers to using several independent microprocessors to process different tasks and jobs at the same time. Multicore processing allows multiple application to run differently at the same time and improve the performance with the extra cores. This is needed for our garage sale app, such being able to take a picture of the object and analyze the information from the previous picture. Failing to support multicore also decrease the reliability and robustness of the application because mistakes could be propagate between software platforms.

## 5. Java

### 5.1 Advantages

Java could run on any platform as long as there is JVM exists on the running machine. Java supports

distributed computing which means that allows several computers to process information and data efficiently. In terms of memory management, Java has two memory storage places which are heap and stack. Heap is responsible for allocation of memory space for objects in Java while stack is responsible for allocation of memory space for execution of threads. Both of them helps organize information and provide ways to restore the data. Java also supports multithreading which allows application to have multiple processes for distributing the tasks by allowing each process to be assigned to one thread. Therefore, each thread can run separately to optimize performance.

### 5.2 Disadvantages

Java indeed consume a lot of memory in terms of management, so its performance is slower compared with C. Even though, Java has the convenient of garbage collector for freeing memory allocation of object, it hurts the application performance. It is because the rest of the threads have to stop in order for the garbage collector to run in order to reclaim back the after-used memory.

## 6. Python

### 6.1 Advantages

Python is considered as one of the interpreted language which executes each line of code from top to bottom. Once the interpreter encounter any error during the process, it will exit the execution and return any possible error and messages to the programmer. Second, Python is dynamically typed, so the programmer do not need to worry about assigning the data type before execution. Third, Python has a huge library that consist of different API and packages that supports for machine learning development.

### 6.2 Disadvantages

Since Python is dynamically typed and interpret code line by line, it runs slower compare with other languages. Also, Python consumer a lot of memory when it comes to building applications due to fact of its simplicity design. Therefore, it is not consider as a good option in mobile computing. It requires a lot of memory which mobile applications has a certain limitation and Python does not process tasks as quick as possible which is the main goal in regards to any applications in mobile devices.

## 7. Conclusion

Based from the above analyzation of among different languages, I think Dart will still be consider the most stainable candidates for developing the garage sale app. Despite of the little delays updates with Flutter, file size problem and lack of third party library support, it should not have a huge problem in regards to the development of the application

# Evaluation Various Languages for a Garage-Sale Buyer App

## 8. Sources

<https://www.cleverism.com/skills-and-tools/dart/>

<https://www.freecodecamp.org/news/what-is-flutter-and-why-you-should-learn-it-in-2020/>

<https://hackernoon.com/why-flutter-uses-dart-dd635a054ebf>

<https://alligator.io/dart/mixins/>

<https://medium.com/codespace69/flutter-interfaces-with-dart-programming-34b3094387fc>

<https://www.w3adda.com/dart-tutorial/dart-abstract-classes>

<https://www.geeksforgeeks.org/just-in-time-compiler/>

<https://dart.dev/guides/libraries>

<https://api.dart.dev/stable/2.7.1/dart-collection/dart-collection-library.html>

<https://hackernoon.com/why-flutter-uses-dart-dd635a054ebf>

<https://habr.com/en/post/455020/>

[https://ocaml.org/learn/tutorials/garbage\\_collection.html](https://ocaml.org/learn/tutorials/garbage_collection.html)

<https://ocamlverse.github.io/content/parallelism.html>

<https://www.geekboots.com/story/parallel-computing-and-its-advantage-and-disadvantage>

<https://help.ableton.com/hc/en-us/articles/209067649-Multi-core-CPU-handling-FAQ>

[https://insights.sei.cmu.edu/sei\\_blog/2017/08/multicore-processing.html](https://insights.sei.cmu.edu/sei_blog/2017/08/multicore-processing.html)

<https://data-flair.training/blogs/pros-and-cons-of-java/>

<https://www.geeksforgeeks.org/disadvantages-of-java-language/>

<https://techvidvan.com/tutorials/python-advantages-and-disadvantages/>