## **BPF.LI/S**



### **KARSTEN SILZ** FEBRUARY 25, 2021



### HOW SHOULD JAVA DEVELOPERS BUILD FRONT-ENUS FUR WEB, MUBILE, AND DESKIUP IUDAY?









## SLDES & VDEDS MORE NFO GET STARTED NEWSLETTER



### **#1: WEB: BUILD ONLY WEB APPLICATION, FROM SCRATCH**

## **#2: MOBILE APPS: WEB APPLICATION EXISTS, BUILD NATIVE IOS & ANDROID APPS FROM SCRATCH**

### **#3: DESKTOP: WEB APPLICATION EXISTS, WHAT TO DO?**

## **3 SCENARIOS**

## **GIVE ADVICE**.

## ... SO WHAT'S MY ANGLE?



### JAVA DEVELOPER FOR 22 YEARS

### **PUT 3 FULL-STACK PROJECTS INTO PRODUCTION SINCE 2017**

### **4TH ONE GOES LIVE THIS SUMMER**

### NOT AFFILIATED, NOT SELLING



## I GIVE YOU OPTIONS FOR YOUR NEXT PROJECT!

APPLY YOUR OWN CRITERIA

### EVALUATE IN YOUR ENVIRONMENT

### MAKE YOUR OWN CHOICES

## (TECHNICAL, LEGAL, ETHICAL, ...)

## WHAT YOU NEED TO DO

APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

EVALUATE IN YOUR ENVIRONMENT

MAKE YOUR OWN CHOICES

### WHAT YOU NEED TO DO

## HOW CAN WE BUILD FRONT-ENDS TODAY?

## HOW SHOULD JAVA DEVELOPERS BUILD FRONT-ENDS TODAY?

## WHAT APPLICATIONS FOR WHICH DEVICES?

DATA ENTRY & RETRIEVAL

**EXPENSE FILING** 

**ISSUE TRACKER** 

### **NO GAMES OR MEDIA APPLICATIONS**



## THESE APPLICATIONS...

## 



## **. ON THESE DEVICES**

### MOBLE E



### EXPENSE FILING OR ISSUE TRACKER



## WHAT APPLICATIONS WHERE?







APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

EVALUATE IN YOUR ENVIRONMENT

MAKE YOUR OWN CHOICES

### WHAT YOU NEED TO DO

EXPENSE FILING OR ISSUE TRACKER



### WHAT APPLICATIONS WHERE?

## HOW CAN WE BUILD FROM ENDS TODAY?

## HOW SHOULD JAVA DEVELOPERS BUILD FRONT-ENDS TODAY?

## WE CAN GO SAFE (STILL HERE IN 5 YEARS)!



### NATIVE



NATIVE











NATIVE



### **ONE CODE BASE O** SLOWER **O** LESS FEATURES **B** JAVASCRIPT **BAD DEBUGGING**

**E 5 CODE BASES E FAST G** ALL FEATURES **E 5 LANGUAGES GOOD DEBUGGING** 

**FAST** 

### **G** ONE CODE BASE

### **G** ALL FEATURES **GOOD DEBUGGING**

## CROSS-PLATFORM



### ANGULAR (GOOGLE)

### **REACT (FACEBOOK)**

### **VUEJS (PERSON: EVAN YOU)**

## **EB FRAMEWORKS**

### FLUTTER (GOOGLE)

### JAVAFX (GLUON)

### **REACT NATIVE (FACEBOOK)**

### XAMARIN (MICROSOFT)

## X-PLATFORM FRAMEWORKS



### **I WORKED WITH ANGULAR & FLUTTER**

### **I ONLY RESEARCHED THE OTHER FRAMEWORKS**

DISCLAIMER

## STATE OF HE ART DECLARATIVE FRONT-ENDS



# 

# APPLE SWIFTU





**Text**("Counter: \(count)")

### Counter: 0



Button (action: { self.count += 1 $}, label: {$ **Text**("Increment")

### er,

### **int)**")

### Counter: 0

Increment



### int)"



Button (action: self.count += 1

### er,

### int)")

2

Counter: 0

Increment



3

### er,

### int)")

Counter: 0

Increment



View VStack (alignment: .center, content: **Text**("Counter: \(count)") .padding() Button (action: { self.count += 1  $}, label: {$ Text ("Increment")



Text("Counter: \(count)")

## 5



**Text**("Counter: \(count)")



## WHAT DID WE JUST SEE?
#### **@State** var count: Int = 0

var **body**: some View { VStack (alignment: .center, content: { Text ("Counter: \(count)") .padding() Button (action: { self.count += 1  $}, label: {$ **Text**("Increment")

## FRONT-END AS CODE WITH BULDER PATTERN



#### **@State** var count: Int = 0

Button (action: { self.count += 1

## COMPONENT STATE **Revents**

#### **@State** var count: Int = 0

**Text**("Counter: \(count)")

## PLATFORM REDRAWS FRONT-END

## FRONT-END AS CODE WITH BUILDER PATTERN

## **COMPONENT STATE & EVENTS**

## PLATFORM REDRAWS FRONT-END

# **DECLARATIVE FRONT-ENDS**

APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

EXPENSE FILING OR ISSUE TRACKER



#### WHAT APPLICATIONS WHERE?

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END



## **.NET MAUI (NOVEMBER 2021)**









SWIFTU

#### SWIFTU

## JETPACK COMPOSE (BETA)





#### **WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS**

#### **CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

### **STATE OF THE ART: DECLARATIVE FRONT-ENDS**

# **SUMMARY: FRONT-ENDS**

APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

**SUMMARY: FRONT-ENDS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END



## **BUILD FROM CAN WE BUILD FROM ENDE TODAY?**

# HOW SHOULD JAVA DEVELOPERS BUILD FRONT-ENDS TODAY?

# JAVA DEVELOPERS WISH LIST



WEB APPS

#### DECLARATIVE

## MOBILE APPS

#### SAFE

#### **DESKTOP APPS**

#### POPULAR

COMPLETE





#### **ALL NATIVE** FEATURES

## **ONE CODE BASE**

#### FAST BUILD, DE-PLOY & DEBUG

# EASIER TO LEARN EASIER TO BUILD, DEPLOY & DEBUG EASIER TO HIRE HARD TO USE "Y" WHEN ALL SAY "X WINS"

# WHY POPULAR?



WEB APPS

#### DECLARATIVE

## MOBILE APPS

#### SAFE

#### **DESKTOP APPS**

#### POPULAR

COMPLETE





#### **ALL NATIVE** FEATURES

## **ONE CODE BASE**

#### FAST BUILD, DE-PLOY & DEBUG

APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

EVALUATE IN YOUR ENVIRONMENT

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

EXPENSE FILING OR ISSUE TRACKER



#### WHAT APPLICATIONS WHERE?

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS
<b>CROSS-PLATFORM</b> FRAMEWORKS: FLUTTER, Javafx, React Native, Xamarin
STATE OF THE ART: DECLARATIVE FRONT-ENDS
SUMMARY, FRONT-FNDS

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END



## **DULD FROM CAN WE DULD FROM ENDE TODA?**

## HOW SHOULD JAVA DEVELOPERS BUILD FRONT-ENDS TODAY?

ANGULAR (GOOGLE)

**REACT (FACEBOOK)** 



## **VUEJS (PERSON: EVAN YOU)**



# **POPULAR: GOOGLE TRENDS**





# **POPULAR: STACK OVERFLOW**





#### **REACT MOST POPULAR, GROWING/FLAT**

## ANGULAR SECOND, SHRINKING

#### VUEJS LAST, GROWING/FLAT

# POPULARITY #1: WEB

# SCORE#1.WEB



WEB APPS

#### DECLARATIVE

## MOBILE APPS

#### SAFE

#### **DESKTOP APPS**

#### POPULAR

COMPLETE





#### **ALL NATIVE** FEATURES

## **ONE CODE BASE**

#### FAST BUILD, DE-PLOY & DEBUG





#### DECLARATIVE REA

#### SAFE REA

## POPULAR REA COMPLETE ANG







#### FAST BUILD, DE-??? PLOY & DEBUG



**REACT WINS** 

## ANGULAR SECOND

#### **VUEJS LAST**





# RANKING #1 - WEB

#### IF YOU ALREADY USE REACT, ANGULAR, OR VUEJS, THEN KEEP T

#### IF YOU DON'T, THEN GO FOR REACT FIRST, THEN ANGULAR, THEN VUE JS

ADVICE#1.WEB

APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

EXPENSE FILING OR ISSUE TRACKER



#### WHAT APPLICATIONS WHERE?

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS
<b>CROSS-PLATFORM</b> FRAMEWORKS: FLUTTER, Javafx, React Native, Xamarin
STATE OF THE ART: DECLARATIVE FRONT-ENDS
SUMMARY, FRONT-FNDS

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, or vue.js, then keep it

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

## HOW CAN WE DULD FROM FROM TODAY?

## HOW SHOULD JAVA DEVELOPERS BUILD FRONT ENDS TODAY?

#### FLUTTER (GOOGLE)

## JAVAFX (GLUON)

## **REACT NATIVE (FACEBOOK)**

# **H2. MOBILE APPS**

## XAMARIN (MICROSOFT)

# **POPULAR: GOOGLE TRENDS**



arin	javafx	
h term	Search term	+

#### Web Search 🔻

	FLUTTE
ACT NATIVE	
and man	



# **POPULAR: STACK OVERFLOW**



a	ti	٧	е
n		-	

## FLUTTER MOST POPULAR, GROWING/FLAT

## **REACT NATIVE SECOND, SHRINKING/GROWING**

## XAMARIN THIRD, SHRINKING/FLAT

### JAVAFX LAST, SHRINKING/FLAT

# POPULARITY #2: MOBILE APPS



**NOT DECLARATIVE** NOT SAFE ENOUGH

## NOT POPULAR ENOUGH



APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

EXPENSE FILING OR ISSUE TRACKER



#### WHAT APPLICATIONS WHERE?

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS
<b>CROSS-PLATFORM</b> FRAMEWORKS: FLUTTER, Javafx, React Native, Xamarin
STATE OF THE ART: DECLARATIVE FRONT-ENDS
SUMMARY, FRONT-FNDS

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END



IF YOU ALREADY USE REACT, ANGULAR, or vue.js, then keep it

IF YOU DON'T, THEN GO FOR REACT FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT
## FLUTTER (GOOGLE)

## JAVAFX (GLUON)

## **REACT NATIVE (FACEBOOK)**

# **H2. MOBILE APPS**

## XAMARIN (MICROSOFT)

## FLUTTER (GOOGLE)

## **REACT NATIVE (FACEBOOK)**





## WHAT IS FLUTTER?

## **"GOOGLE'S UI TOOLKIT FOR** BUILDING BEAUTIFUL, NATIVELY **COMPILED APPLICATIONS FOR** MOBILE. WEB. AND DESKTOP FROM A SINGLE CODEBASE."

## "GOOGLE'S UI TOOLKIT FOR BUILDING BEAUTIFUL. NATIVELY **COMPILED APPLICATIONS FOR** MOBILE WEB. AND DESKTOP FROM **ASINGLE CODEBASE."**

## FLUTTER PROTOTYPE SUMMER 2019

### FLUTTER PROJECT FOR IOS & ANDROID **SINCE SEPTEMBER 2020**



# **ONLY GOVER** FLUTTER ON MOBILE IN THIS TALK

## **O** JAVASCRIPT **O** SLOWER RENDERING **G** NATIVE UI ELEMENTS **E NO DESKTOP SUPPORT** $\bigcirc$ REACT + REACT NATIVE $\bigcirc$ ONE CODEBASE **REACT NATIVE** FLUTTER

## **DART G FASTER RENDERING O EMULATED UI ELEMENTS G** ALL DEVICES

APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

### WHAT YOU NEED TO DO

EXPENSE FILING OR ISSUE TRACKER



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

### **SUMMARY: FRONT-ENDS**

**JAVASCRIPT SLOWER RENDERING** NATIVE UI ELEMENTS NO DESKTOP SUPPORT
ALL DEVICES **REACT + REACT NATIVE ONE CODEBASE** 

**DART FASTER RENDERING** 😑 EMULATED UI ELEMENTS 🛛



FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END



IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

### XAMARIN & JAVAFX: OUT

# WHAT FRONT-ENDS CAN WE BULD WITH FLUTTER?



Home Series Films New & Popular My List

### Action





WORL 🥥 2021 NEW EPISODES

SNOWPIERCER NEW EPISODE WEEKLY

IOTE

RTENIE

8 Seasons 😳 2019 Suspenseful . Mystery 2 Seasons 🧕 2021 Dystopian • Intimate

### **US TV Action & Adventure**



Exciting + Sci-Fi TV

COBRA KAI

3 Seasons 🔘 2021 Rousing . Heartfelt

4 Seasons 🧕 2018



3 Seasons 🥥 2018





SPACE SWEEPERS

0 2021 Exciting \* Sci-Fi Drama







0 2019

0 2020 Action Thriller \* Action & Adventure

# 

### Q 🗄 🧳 🚺 -



1h 34m 🧔 2018 Dystopian - Violent



I RIBEZ 1 Season 🥥 2021

Dystopian • Dark





2 Seasons 🔘 2019 Gritty Thriller







### **Netflix Originals**

1.

MA



## **GOOGLE'S DESIGN LANGUAGE**

## **CONSIDERED NATIVE UI ON ANDROID**



#2: MATERIAL DESIGN

Dect code
Doct code
Doct code
Post code
CVV2





	_		×	 15:41 <b>-</b>		.ıl 🗢 🛙
			~ ?	Cancel	Edit Profile	Sa
Open ▼ Edit	Select all	ne				
History		ection		Salutation		Μ
n	Select			Karsten		
<b>,</b> ,	Search Loca	Disk (C:	)	middle name		
	Size			Test		
:	5126					
folder						
older						
older						
older						
older						
older						
older						
older						
older						
older						
		0 KB				
Document		1 KB				
				ž	• • • • • • • •	2
				Tasks	Team	My Data

# 

 $\otimes$ 

 $\bigotimes$ 

# REALY MOBILE, WEB & DESKTOP FROM ONE CODEBASE WITH FLUTER?







### ALPHA **ALPHA ALPHA** SINCE SINCE SINCE JUL 2020 **DEC 2019 SEP 2020**





BETA SINCE **DEC 2019** 

### **DEC 2018**



### 



# WHAT FRONT-ENDS CAN WE BULD WITH FLUTTER?

### OWN YES YES BRAND MATERIAL YES YES

NATIVE NO NO





**IOS & ANDROID: YES** 

## WEB: BETA, NO NATIVE UI

## DESKTOP: ALPHA, NO NATIVE UI

# FLUTTER: ONE CODEBASE?

APPLY YOUR OWN CRITERIA (TECHNICAL, LEGAL, ETHICAL, ...)

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

### **SUMMARY: FRONT-ENDS**

**JAVASCRIPT SLOWER RENDERING H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT
ALL DEVICES **REACT + REACT NATIVE ONE CODEBASE** 

**DART FASTER RENDERING** 😑 EMULATED UI ELEMENTS 🛛

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END



IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

### XAMARIN & JAVAFX: OUT



### Flutter Engage

Join the Flutter community around the world for this special online event! There will be opportunities to engage, innovate, and collaborate with the global community.

March 3, 2021 9:30AM - 1:30PM PST

Sign up for updates

Add to Calendar

# CHANGES ON 3 MARCH?



# CANTWEUSE JAVA INSTEAD OF DART?

class MyClass extends AnotherClass { String myString; int myInt;

String sayHello (String name) { var feedback = "Hello, " + name; return feedback;



### List<String> myList = new List<String>();



class MyClass extends AnotherClass { String myString; int myInt;

String sayHello(String name) { var feedback = "Hello, " + name; return feedback;



### List<String> myList = new LinkedList<>();



## GOOGLE BUILT IT ORIGINALLY FOR BROWSER "SIMPLIFIED JAVA", STATICALLY TYPED

## **USEFUL CHANGES FOR BUILDING FRONT-ENDS**

## VM DURING DEV, NATIVE CODE IN PRODUCTION

# WHAT IS DART?

**APPLY YOUR OWN CRITERIA** (TECHNICAL, LEGAL, ETHICAL, ...)

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS **REACT + REACT NATIVE ONE CODEBASE** 

**DART FASTER RENDERING** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

EXPENSE FILING OR ISSUE TRACKER



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR REACT FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

### XAMARIN & JAVAFX: OUT

## STAND-ALONE APP

## **AS SCREENS IN NATIVE APP**



## MOBILE BUILD TIMES ARE LONG HOT RELOAD SHOWS CHANGES IMMEDIATELY

HOT RESTART STARTS APP ANEW

FASTER THAN NATIVE IOS/ANDROID

# FAST BUILD & DEPLOY!

# HOT RELOAD IS FUN & KEEPS THE FLOW!

## HAVE BARELY USED THEM

## FOR APP TUNING

## FLUTTER WEB APP







## PLUGINS ADD FUNCTIONALITY

**MOST OPEN-SOURCE** 

## **GREAT PLUGIN PORTAL**



PLUGINS

## THROUGH PLUGINS

## SHOW NATIVE SCREENS IN FLUTTER

### TALK TO NATIVE CODE THROUGH CHANNELS (PUBLISH & SUBSCRIBE) OR C API

# NATIVE INTEGRATION

# PLUGIN HAS BUGS OR NOT MAINTAINED? **1. FORK IT ON GITHUB** 2. MAKE CHANGES IN FORK **3. USE FORK GIT URL IN FLUTTER BUILD FILE** FXIG PLUGINS

# FAST CODE, BUILD & DEPLOY WITH DART & TOOLS PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION **OPEN-SOURCE AT GITHUB: SOURCE, ISSUES**

# FLUTTER: GOOD PARTS
**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER.** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

EXPENSE FILING OR ISSUE TRACKER



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

#### **ADVICE #1: WEB**

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS



# MY FLUTTER ISSUES

# WILL GOOGLE KILL FLUTTER?



Search



June 2022

"Off with their heads!" in over 1 year, Google Chrome Apps were hosted or packaged web applications that ran on the Google Chrome browser. It will be over 11 years old.



Join a bunch of others and follow @killedbygoogle on Twitter.





December 2021

Biting the big one in 10 months,

AngularJS was a JavaScript open-source front-end web framework based on MVC pattern using a dependency injection technique. It will be about 11 years old.



2021



Another one bites the dust in 4 months, Google Hangouts was a communication platform which included messages, video chat, and VOIP features. Execution scheduled "first half 2021". It will be over 8 years old.

All (225)





 $\sim$ 

January 2022



Turning to ashes in 10 months, Android Things was an Android-based embedded operating system (originally named Brillo) aimed to run on Internet of Things (IoT) devices. It will be over 6 years old.



#### June 2021

#### Expeditions

To be exterminated in 4 months, Expeditions is a program for providing virtual reality experiences to school classrooms through Google Cardboard viewers, allowing educators to take their students on virtual field trips. It will be almost 6 years old.





## BAD: COMPETING EFFORTS AT GOOGLE GOOD: POPULAR OUTSIDE OF GOOGLE GOOD: FLUTTER UI IN PROJECT FUCHSIA

# NOBODY KNOWS!

# TWO WAYS TO Arrange Elements...



### COLUMN: VERTICAL LIST



### ROW: HORIZONTAL LIST

## PHONE ROTATION OR BIG PHONES/TABLETS? FLUTTER: ONLY SCREEN DIMENSIONS **I USE PLUGIN FOR "BOOTSTRAP RESPONSIVE GRID"**

# BUT NOT RESPONSIVE!

## IOS UI ELEMENTS MISSING, TOO MUCH MANUAL ASSEMBLY **USE PLUGINS TO ADD MISSING ELEMENTS & ABSTRACT IOS/ANDROID**

### THEN BUILD YOU OWN ABSTRACTIONS ON TOP

## NATIVE UI TOO HARD

## WILL GOOGLE KILL FLUTTER?

## **NO RESPONSIVE DESIGN BUILT-IN**

### NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

IF YOU USE MATERIAL DESIGN ON DESKTOP AND USE FLUTTER: EVALUATE FLUTTER FOR DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#3: DESKTOP APPS: YES – USE WEB APPLICATION DO | FOLLOW MY OWN ADVICE?** 

**IOS & ANDROID APPS** 

**REACT FOR SIDE PROJECT & NEW PROJECTS** 

**#2: MOBILE APPS: YES – FLUTTER FOR NATIVE** 

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS **#1: JUST WEB: YES – USE ANGULAR NOW, LEARN** 

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH



XAMARIN & JAVAFX: OUT

# SCORE #2: MOBILE APPS



WEBAPPS REA

## DECLARATIVE - MOBILE APPS -

### SAFE REA

## POPULAR FLU





FAST FLU

# ALL NATIVE ??? FEATURES

## DESKTOP APPS FLU ONE CODE BASE FLU



### FAST BUILD, DE-PLOY & DEBUG



### FLUTTER WINS

## **REACT NATIVE SECOND**

## JAVAFX AND XAMARIN OUT

# RANKING #2: MOBILE APPS



OR FLUTTER, THEN KEEP T

### IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

# **IF YOU ALREADY USE REACT NATIVE**

# ADVICE #2: MOBILE APPS

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **SLOWER RENDERING H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING EMULATED UI ELEMENTS** 

**REACT NATIVE FLUTTER** 

EXPENSE FILING OR ISSUE TRACKER



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER, THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST. THEN ANGULAR. THEN VUE.JS

#### **ADVICE #1: WEB**

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS



## HOW CAN WE DULD FROM FROM TODAY?

## HOW SHOULD JAVA DEVELOPERS BUILD FRONT ENDS TODAY?

# **H3. DESKTOP**

## NO GOOD CROSS-PLATFORM OPTION FOR DESKTOP BIG CHANGES IN WINDOWS & MAC WEB: GOOD ENOUGH ON DESKTOP

## THE WATING GAME

### **KEEP YOUR WEB APPLICATION FOR DESKTOP**

### IF YOU USE MATERIAL DESIGN ON DESKTOP AND USE FLUTTER: EVALUATE FLUTTER FOR DESKTOP IN 2022



**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING EMULATED UI ELEMENTS** 

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER, THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

IF YOU USE MATERIAL DESIGN ON DESKTOP AND USE FLUTTER: EVALUATE FLUTTER FOR DESKTOP IN 2022

**ADVICE #3: DESKTOP** 



## HOW CAN WE DULD FROM FROM TODAY?

## HOW SHOULD JAVA DEVELOPERS DUILD FRONT ENDS TODAY?

## #1: JUST WEB: YES - ANGULAR NOW, LEARN REACT **#2: MOBILE APPS: YES - FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES - USE WEB APPLICATION**

# DO FOLLOW MY OWN ADVICE?



**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 

**DO | FOLLOW MY OWN ADVICE?** 







# WRAP-UP

## **#1: WEB: BUILD ONLY WEB APPLICATION, FROM SCRATCH**

# **#2: MOBILE APPS: WEB APPLICATION EXISTS, BUILD NATIVE IOS & ANDROID APPS FROM SCRATCH**

### **#3: DESKTOP: WEB APPLICATION EXISTS, WHAT TO DO?**

# **3 SCENARIOS**

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 

**DO | FOLLOW MY OWN ADVICE?** 







APPLY YOUR OWN CRITERIA

### EVALUATE IN YOUR ENVIRONMENT

### MAKE YOUR OWN CHOICES

# (TECHNICAL, LEGAL, ETHICAL, ...)

## WHAT YOU NEED TO DO

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 

**DO | FOLLOW MY OWN ADVICE?** 







### EXPENSE FILING OR ISSUE TRACKER



# WHAT APPLICATIONS WHERE?





**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 

**DO | FOLLOW MY OWN ADVICE?** 







## FRONT-END AS CODE WITH BUILDER PATTERN

## **COMPONENT STATE & EVENTS**

## PLATFORM REDRAWS FRONT-END

# **DECLARATIVE FRONT-ENDS**

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 

**DO | FOLLOW MY OWN ADVICE?** 







### **WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS**

### **CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

## **STATE OF THE ART: DECLARATIVE FRONT-ENDS**

# **SUMMARY: FRONT-ENDS**

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 

**DO | FOLLOW MY OWN ADVICE?** 









WEB APPS

### DECLARATIVE

## MOBILE APPS

### SAFE

### **DESKTOP APPS**

### POPULAR

COMPLETE





### **ALL NATIVE** FEATURES

## **ONE CODE BASE**

### FAST BUILD, DE-PLOY & DEBUG
**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







### IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP T

### IF YOU DON'T, THEN GO FOR REACT FIRST, THEN ANGULAR, THEN VUE.JS

## ADVICE #1. WEB

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







**NOT DECLARATIVE** NOT SAFE ENOUGH

### NOT POPULAR ENOUGH



**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







### **O** JAVASCRIPT **O** SLOWER RENDERING **G** NATIVE UI ELEMENTS **E NO DESKTOP SUPPORT** $\bigcirc$ REACT + REACT NATIVE $\bigcirc$ ONE CODEBASE **REACT NATIVE** FLUTTER

## **DART G FASTER RENDERING O EMULATED UI ELEMENTS G** ALL DEVICES

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







**IOS & ANDROID: YES** 

### WEB: BETA, NO NATIVE UI

### DESKTOP: ALPHA, NO NATIVE UI

## FLUTTER: ONE CODEBASE?

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







### GOOGLE BUILT IT ORIGINALLY FOR BROWSER "SIMPLIFIED JAVA", STATICALLY TYPED

### **USEFUL CHANGES FOR BUILDING FRONT-ENDS**

### VM DURING DEV, NATIVE CODE IN PRODUCTION

## WHAT IS DART?

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







## FAST CODE, BUILD & DEPLOY WITH DART & TOOLS PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION **OPEN-SOURCE AT GITHUB: SOURCE, ISSUES**

## FLUTTER: GOOD PARTS

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







### WILL GOOGLE KILL FLUTTER?

### **NO RESPONSIVE DESIGN BUILT-IN**

### NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







OR FLUTTER, THEN KEEP T

### IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

## **IF YOU ALREADY USE REACT NATIVE**

## ADVICE #2: MOBILE APPS

**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







### **KEEP YOUR WEB APPLICATION FOR DESKTOP**

### IF YOU USE MATERIAL DESIGN ON DESKTOP AND USE FLUTTER: EVALUATE FLUTTER FOR DESKTOP IN 2022



**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







### #1: JUST WEB: YES - ANGULAR NOW, LEARN REACT **#2: MOBILE APPS: YES - FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES - USE WEB APPLICATION**



**EVALUATE IN YOUR ENVIRONMENT** 

MAKE YOUR OWN CHOICES

#### WHAT YOU NEED TO DO

WEB FRAMEWORKS: ANGULAR, REACT, VUE.JS

**CROSS-PLATFORM FRAMEWORKS: FLUTTER,** JAVAFX, REACT NATIVE, XAMARIN

STATE OF THE ART: DECLARATIVE FRONT-ENDS

#### **SUMMARY: FRONT-ENDS**

**—** JAVASCRIPT **—** SLOWER RENDERING **H**NATIVE UI ELEMENTS NO DESKTOP SUPPORT 
ALL DEVICES  $\bigcirc$  REACT + REACT NATIVE  $\bigcirc$  ONE CODEBASE

**DART FASTER RENDERING —** EMULATED UI ELEMENTS

**REACT NATIVE FLUTTER** 

**EXPENSE FILING OR ISSUE TRACKER** 



#### WHAT APPLICATIONS WHERE?

JAVA	WEB APPS	FAST
DECLARATIVE	MOBILE APPS	ALL NATIVE Features
SAFE	DESKTOP APPS	ONE CODE BASE
POPULAR	COMPLETE	FAST BUILD, DE- Ploy & Debug

IOS & ANDROID: YES

WEB: BETA, NO NATIVE UI

**DESKTOP:** ALPHA, NO NATIVE UI

#### FLUTTER: **ONE** CODEBASE?

WILL GOOGLE KILL FLUTTER?

**NO RESPONSIVE DESIGN BUILT-IN** 

NATIVE UI TOO HARD

FLUTTER: BAD PARTS

**IF YOU ALREADY USE REACT NATIVE** OR FLUTTER. THEN KEEP IT

IF YOU DON'T, THEN GO FOR FLUTTER FIRST, THEN REACT NATIVE

**ADVICE #2: MOBILE APPS** 

FRONT-END AS CODE WITH BUILDER PATTERN

**COMPONENT STATE & EVENTS** 

PLATFORM REDRAWS FRONT-END

#### **DECLARATIVE FRONT-ENDS**

IF YOU ALREADY USE REACT, ANGULAR, OR VUE.JS, THEN KEEP IT

IF YOU DON'T, THEN GO FOR **REACT** FIRST, THEN ANGULAR, THEN VUE.JS

**ADVICE #1: WEB** 

**GOOGLE BUILT IT ORIGINALLY FOR BROWSER** "SIMPLIFIED JAVA", STATICALLY TYPED **USEFUL CHANGES FOR BUILDING FRONT-ENDS** VM DURING DEV, NATIVE CODE IN PRODUCTION WHAT IS DART?

**NOT DECLARATIVE** NOT POPULAR ENOUGH NOT SAFE ENOUGH

#### XAMARIN & JAVAFX: OUT

FAST CODE, BUILD & DEPLOY WITH DART & TOOLS **PLUGINS ADD FUNCTIONALITY & PLUG HOLES EASY NATIVE INTEGRATION OPEN-SOURCE** AT GITHUB: SOURCE, ISSUES FLUTTER: GOOD PARTS

**KEEP YOUR WEB APPLICATION FOR DESKTOP** 

ATERIAL DESIGN ON D IF YOU USE M **R: EVALUATE FLUTTER FOR AND USE FLUT** DESKTOP IN 2022

**ADVICE #3: DESKTOP** 

**#1: JUST WEB: YES – ANGULAR NOW, LEARN REACT #2: MOBILE APPS: YES – FLUTTER FOR IOS & ANDROID #3: DESKTOP APPS: YES – USE WEB APPLICATION** 







## 

## **OCTOBER 2021 CONTRACTOR OR FIXED-TERM EMPLOYEE**

# MILTON KEYNES, LONDON OR REMOTE

## I'M LOOKING FOR WORK





## SLDES & VDEDS MORE NFO GET STARTED NEWSLETTER

