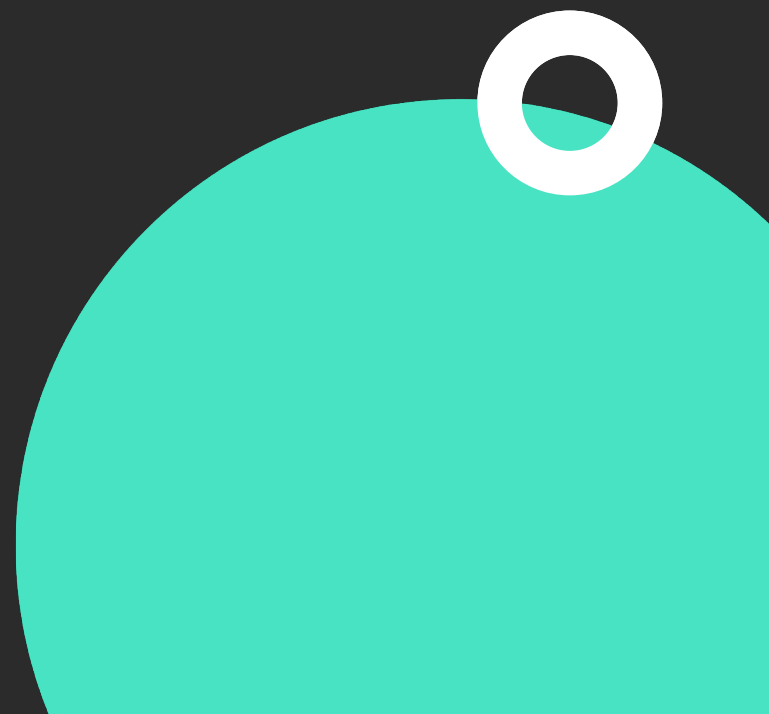# Configuring Build Types and Product Flavors in Android

Kaushal Dhruw

@drulabs

in/kaushal-dhruw
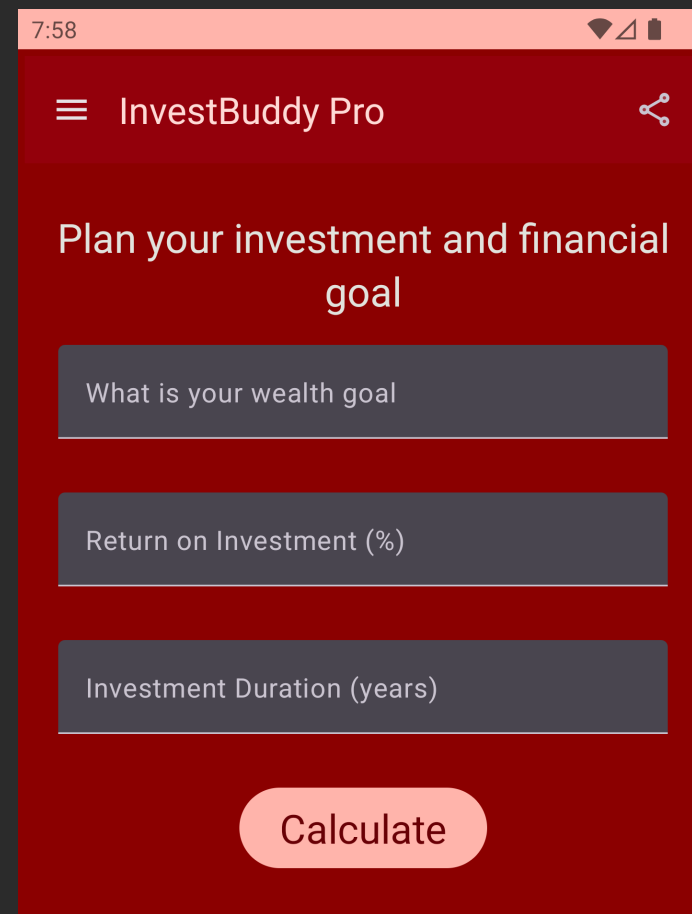
Overview

- Why Product Flavors

- Explore Built types, Product flavors and Build variants

- Explore the starter project with white labelled demo app

- Create new Build Type, Product Flavor and flavorDimension

- Create Build Variant specific color scheme and theming

- Coding **Challenge**

- Add Crashlytics to individual Flavors

- When to use and when to avoid Product Flavors

Why Product Flavors?

- Multi-Environment support

- Branding and Theming

- Build free and premium versions
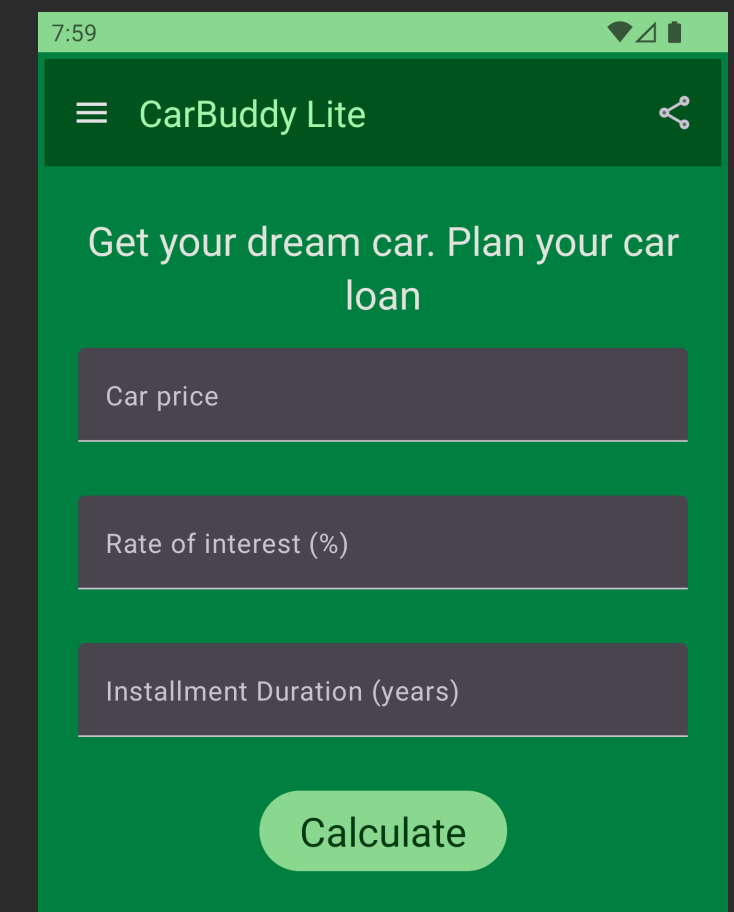
Build Types

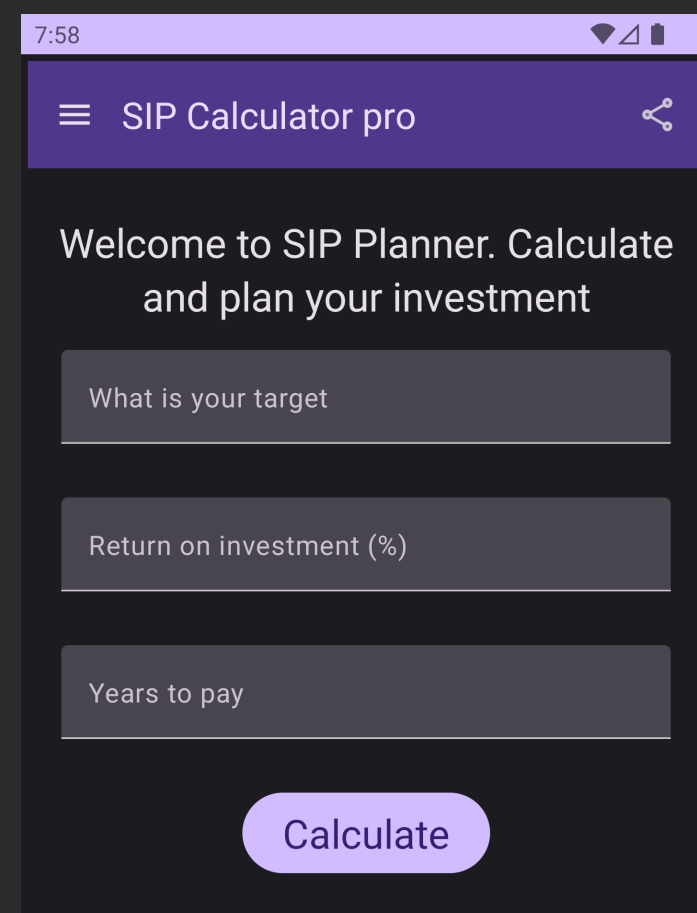- Defines certain attributes to build the app

- "debug" and "release" build types are defined by default and can be configured

- "debug" build type uses debug key to sign the app

- "release" build type should be signed with the release key for distribution

Debug   Alpha   Beta   Release

Build Types

- Refer to the existing build type type using getByName("")

- Create new build types

- New build types can be initialized with existing ones using initWith.

- initWith copies all properties from the given build type

```
buildTypes {
    getByName("debug") {
        isMinifyEnabled = false
    }
    getByName("release") {
        isMinifyEnabled = true
        proguardFiles(...)
    }
    create("alpha") {
        initWith(getByName("debug"))
        ...
    }
    create("beta") {
        initWith(getByName("release"))
        ...
    }
}
```

Product Flavors and Dimensions

- Product flavors represent a versions of an app

- All product flavors share a common source

- Resources and code can be customized for each product flavor

- All product flavors must belong to a named flavor dimension

- You can define flavor specific gradle dependency.

Defining Product Flavors

- We've been using product flavors all along

- defaultConfig belongs to ProductFlavor class

- Flavors can override any of the attributes of defaultConfig like applicationId

- Product flavors are combined with Build Types to create build variants

```
defaultConfig {
    applicationId = "com.app.default"
    targetSdk = 34
    versionName = "3.2.1"
    ...
}
flavorDimensions += "color"
productFlavors {
    create("red") {
        dimension = "color"
        applicationId = "com.app.red"
        versionName = "1.0.8"
    }
    create("green") {
        dimension = "color"
        applicationId = "com.app.green"
        versionName = "2.1.7"
    }
}
```

droidcon
academy

Build Variant

Built Types

Product Flavor Dimension: Color

Product Flavor Dimension: Opacity

debug

release

Blue

Red

Alpha50

Alpha80

Built Variants →

BlueAlpha50Debug

BlueAlpha50Release

RedAlpha50Debug

RedAlpha50Release

BlueAlpha80Debug

BlueAlpha80Release

RedAlpha80Debug

RedAlpha80Release

droidcon academy
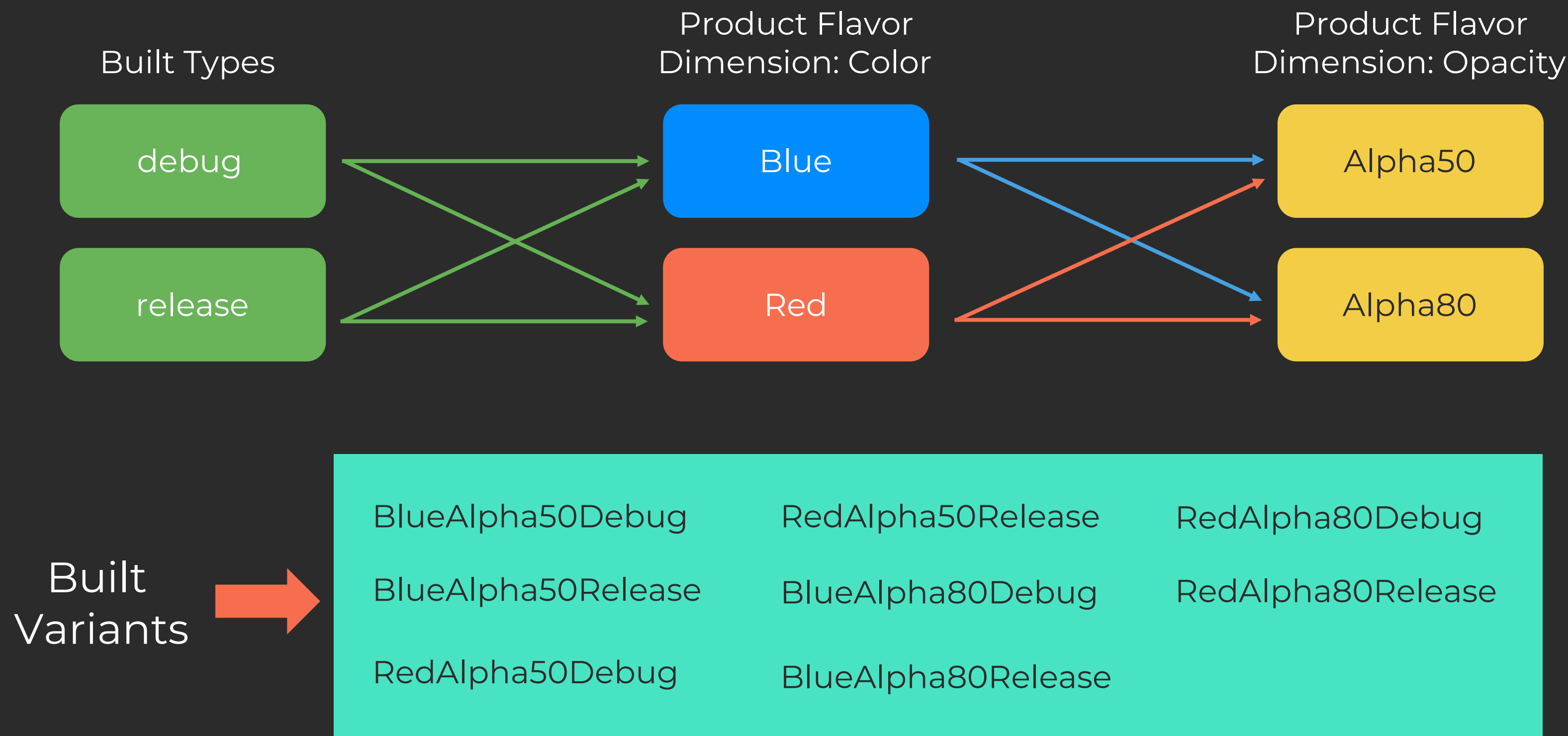
Demo

Wealthify: A white labelled app that calculates a Simple Interest based monthly premium for a target amount. Saves a few historical calculations.

- Explore existing product flavors
- Create dimension and product flavors
- Filter out unwanted flavors
- Flavor specific theming and customization
- Crashlytics integration

7:30

☰ WealthBuddy ⌁

Enter the values as suggested below to get started

Enter Amount
40000

Rate of Interest (%)
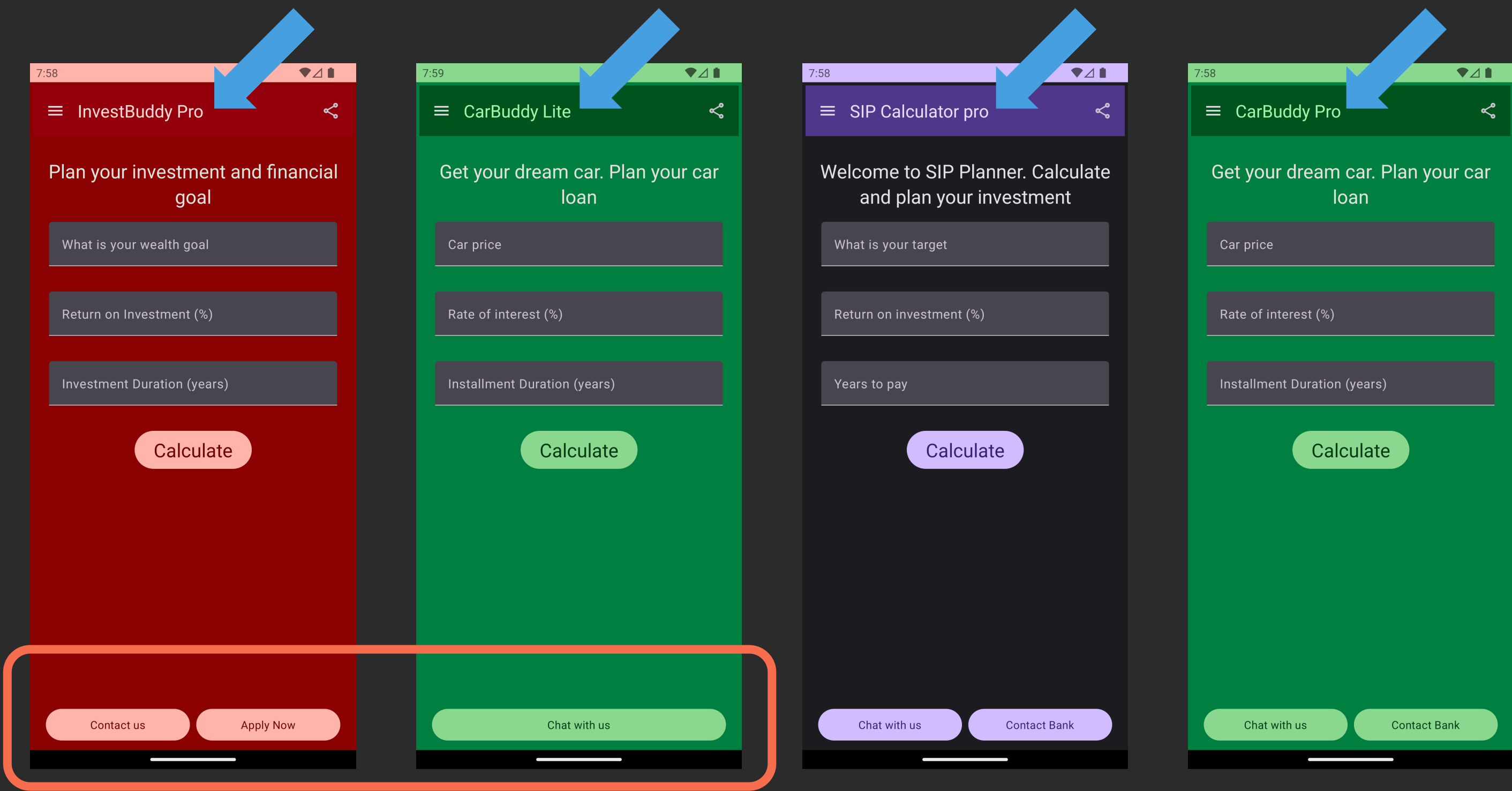12

Tenure in years
4

Calculate

Amount = [27,027.03], Maturity amount = [40,000.00]
Pay 563.06 monthly for 48 months

Connect with us   Apply

Branding the White Labelled App

Coding Challenge ⚙️

- Create a new Product flavor called "investBuddy" under "customer" dimension

- Make sure to set the "isInvestment" build field to "true"

- You can get resources from the helper-files section of the repository

- Configure the apply and connect URLs to any new URLs

- Name the free and premium versions as "InvestBuddy Lite" and "InvestBuddy Pro" respectively.

- (optional) Add signing config for InvestBuddy variants

droidcon academy

Coding Challenge Solution 🤖

When to Use and When NOT to

- Your app needs multiple variants

- Custom configuration like API endpoints

- Different testing environments

- Targeting different regions or market

- Your app is straightforward and very basic

- Versions differ too much in UI and implementation

- Limited audience

droidcon academy

Summary

- Explore build types, product flavors and build variants

- Create different versions of the white labelled demo app

- Create build type and product flavors

- Add flavor specific themes and resources

- Filter product flavors

- Extract fields into BuildConfig wherever applicable

- Integrate Firebase to various flavors

- When to use and when to avoid product flavors

droidcon academy

Thank You!!

Kaushal Dhruw

🐦 @drulabs

in in/kaushal-dhruw