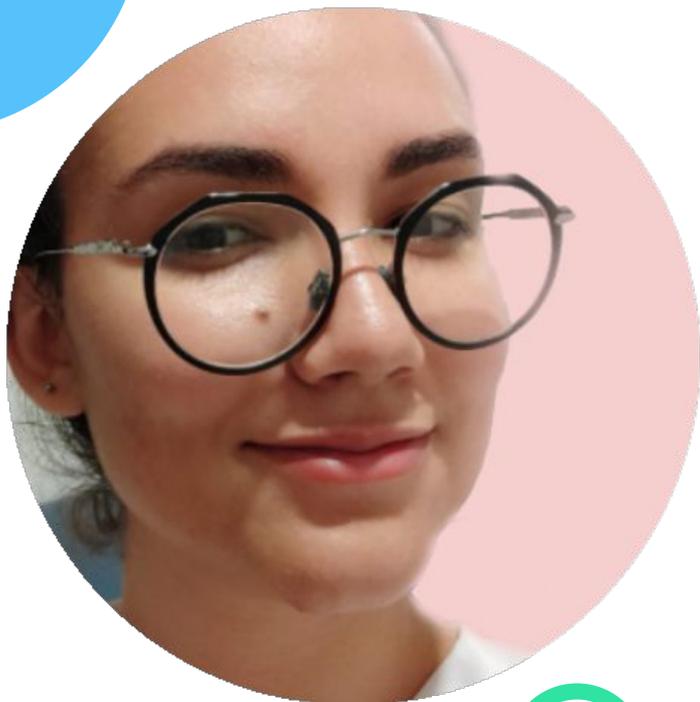


Themes & Styles

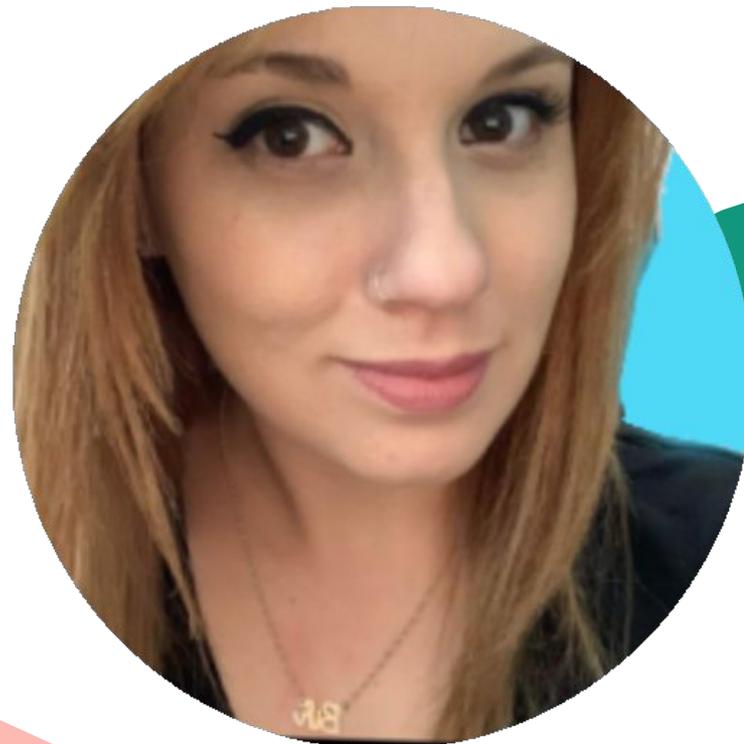
Conheça os aliados na construção dos SuperApps globais e multimarca da **Natura &Co**



QUEM SOMOS

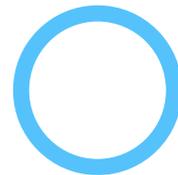


Marcella Souza - Deva Mobile
@Thoughtworks
LinkedIn: marcellasouza



Bianca Letti - Engineering Manager
@Natura
LinkedIn: biancaletti

HOJE VAMOS FALAR DE...



01

NOSSO CONTEXTO

O que nos motivou a criar a solução que vamos compartilhar

02

E AÍ ANDROID

Ferramentas do Android Nativo que exploramos

03

APRENDIZADOS

O que deu certo e o que deu errado em nossas tentativas

NOSSO CONTEXTO

2016

Aesop

Finalizamos a
aquisição da
australiana
Aesop

2017

**The Body
Shop**

Finalizamos a
aquisição da TBS

2020

Avon

Finalizamos a
aquisição da
Avon

...

E agora?

O que fazer
sendo o 4o maior
grupo de beleza
do Mundo?

Contexto: o PROBLEMA?



01

Múltiplas aplicações

Já éramos multi canal, multi país - e passamos a ser multi marca.

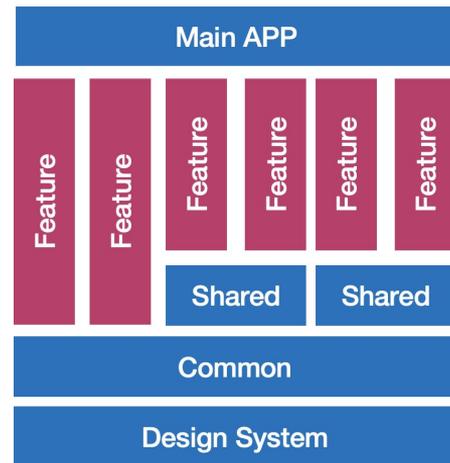
02

Frontends?

O desafio era conseguir ajustar a asa do avião com ele voando.

Monolito x Módulos

Monolito



O QUE é o DESIGN SYSTEM DA NATURA?



03

Um produto de produtos

O Design não é escalado com facilidade. E como a gente resolve?

04

Bibliotecas

Desacoplar, abstrair e servir. A tecnologia em prol de alguma coisa.

UM POUQUINHO DE NATURA

Venda Direta

B2C

Varejo



NOSSO CAMINHO

Agente transformador

Código aberto: queremos ser mais um agente de fomento para a comunidade

Cultura da contribuição

A biblioteca é produto do trabalho conjunto das squads



NOSSO CAMINHO

Isolamento de funções

Parte visual comum das apps fica separado das outras funcionalidades de negócio

Consistência visual

Produtos alinhados com estratégia visual das marcas mais rapidamente

E AÍ ANDROID? AJUDA 'NÓIS'!

Biblioteca Android

- Baseada no *Material Components*
- Temas
- Estilos
- Componentes customizados





TEMAS | COMO APLICAMOS

```
<style name="Theme.Natura.Light" parent="Theme.MaterialComponents.Light">
  <item name="colorPrimary">#F4AB34</item>
  <item name="colorOnPrimary">#333333</item>
  <item name="colorPrimaryLight">#FEFDE8</item>
  <item name="colorOnPrimaryLight">#333333</item>

  <item name="buttonStyle">@style/Widget.DS.Button.V23</item>
```

```
<androidx.constraintlayout.widget.ConstraintLayout
  android:id="@+id/colorPrimaryContainer"
  android:layout_width="match_parent"

  android:background="?colorPrimary">
```



TEMAS | COMO APLICAMOS

Para cada marca que nossa biblioteca suporta, associamos propriedades base dos nossos *layouts* (*design tokens*) além da estilização dos componentes.

The Body Shop



?colorPrimary



Botão

(fundo branco, borda com cor primária ...)

Natura



?colorPrimary



BOTÃO

(fundo cinza claro, borda com cor primária, texto sem itálico, todas letras maiúsculas ...)



TEMAS | COMO APLICAMOS

O temas das marcas são selecionado a partir do flavor indicado no build.

```
4      natura {  
5          ...  
6          dimension "default"  
7          manifestPlaceholders = [  
8              appTheme: "@style/AppTheme.Natura.NoActionBar"  
9          ]  
10     }  
11     theBodyShop {  
12         ...  
13         manifestPlaceholders = [  
14             appTheme: "@style/AppTheme.BodyShop.NoActionBar"  
15         ]  
16     }  
17     avon {  
18         ...  
19         manifestPlaceholders = [  
20             appTheme: "@style/AppTheme.Avon.NoActionBar"  
21         ]  
22     }
```



TEMAS | COMO APLICAMOS

Podemos aplicar esse tema de diferentes formas...

- Manifest: tag *application* ou tag *activity*
- View: atributo *theme* ou *setTheme()*
- ...

```
<manifest
  xmlns:tools="http://schemas.android.com/tools"
  xmlns:android="http://schemas.android.com/apk/res/android">
  <application tools:ignore="ExtraText">
    ...

    <activity
      android:name="ExempleActivity"
      android:theme="{appTheme}" />
  </application>
</manifest>
```



ESTILOS | COMO APLICAMOS

Usamos os estilos para agrupar os atributos das views.

Em algumas situações customizamos estilos que o Material oferece.

```
<!-- BUTTON STYLES -->
<style name="Widget.DS.Button.V23" parent="Widget.MaterialComponents.Button">
  <item name="rippleColor">@color/button_ripple_color_v23</item>
  <item name="android:textColor">@color/button_contained_text_color_selector_v23</item>
  <item name="backgroundTint">@color/button_background_color_selector_v23</item>
</style>
```

```
<style name="Widget.DS.Button.Outlined.V23" parent="Widget.MaterialComponents.Button.OutlinedButton">
  <item name="android:textColor">@color/button_text_color_selector_v23</item>
  <item name="strokeColor">@color/stroke_color_selector_v23</item>
  <item name="rippleColor">@color/button_ripple_color_v23</item>
</style>
```



ESTILOS | COMO APLICAMOS

Usamos os estilos para criar variantes dos componentes

```
<Button
  android:id="@+id/textButton"
  style="?textButton"
  android:layout_width="wrap_content"
  android:minWidth="?sizeHuge"
  android:layout_height="wrap_content"
  android:text="text" />

<Button
  android:id="@+id/outlinedButton"
  style="?outlinedButton"
  android:layout_width="wrap_content"
  android:minWidth="?sizeHuge"
  android:layout_height="wrap_content"
  android:text="outlined" />
```

Botão com
Preenchimento

Botão com Borda

Botão com Texto



COMPONENTES | COMO APLICAMOS

Criamos uma *custom view* sempre que precisamos de algo que não é suportado nativamente ou que simplifica a experiência de desenvolvimento quando componentizado.

```
class AppBar(context: Context, attrs: AttributeSet) : Toolbar(context, attrs) {
```

```
<com.natura.android.appbar.AppBar  
    android:id="@+id/appBar"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    app:showLogo="true"/>
```



NOSSO ALIADOS NO ANDROID



Temas

Conjunto de estilos aplicados à um contexto



Estilos

Mapeamento de valores aos atributos das views



Componentes

Item da interface customizado e replicável

Essas são as três principais ferramentas nativas que usamos para criar nossa biblioteca

O QUE APRENDEMOS ATÉ AQUI?



01

GESTÃO DA BIBLIOTECA

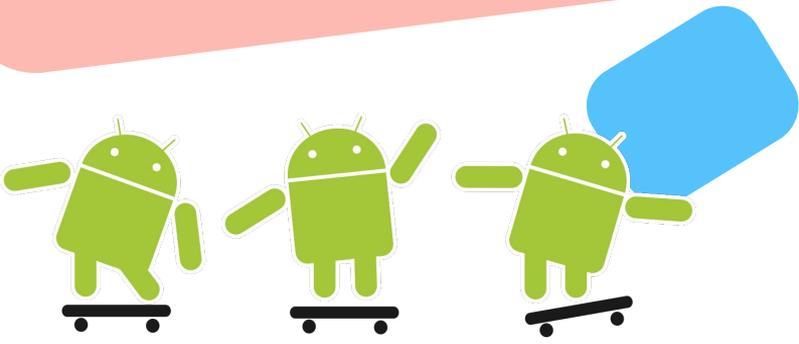
Usar e abusar de automações para deixar a gestão mais simples

02

MATERIAL DESIGN

Ponderar quando customizar x quando criar um componente próprio

O QUE APRENDEMOS ATÉ AQUI?



03

RETROCOMPATIBILIDADE

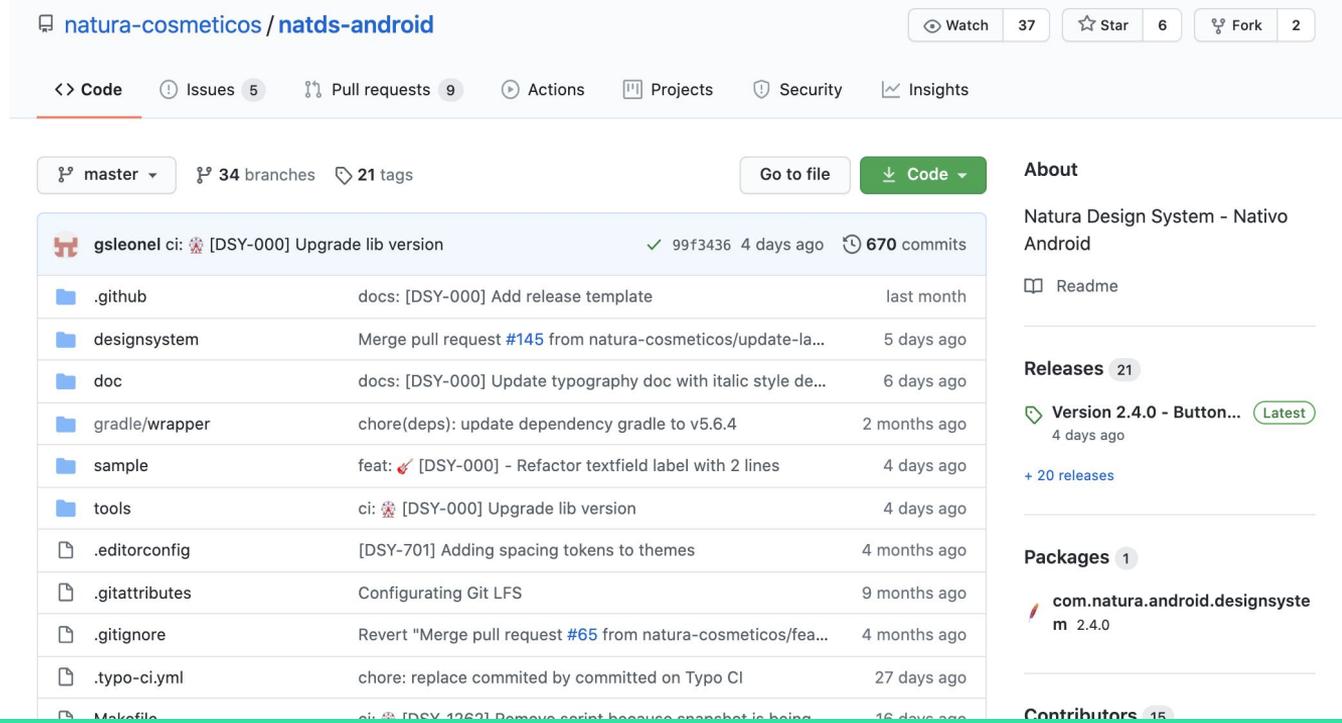
Testar com cuidado as customizações

04

CENTRADO NAS USUÁRIAS

As desenvolvedoras também são usuárias da biblioteca. Pensar na exp. dessas pessoas é importantíssimo

QUER SABER MAIS? ACESSE!



natura-cosmeticos / natds-android

Watch 37 Star 6 Fork 2

<> Code 5 Issues 5 Pull requests 9 Actions Projects Security Insights

master 34 branches 21 tags Go to file Code

Commit	Message	Time
gsleone1 ci: [DSY-000] Upgrade lib version	99f3436 4 days ago	670 commits
.github	docs: [DSY-000] Add release template	last month
designsystem	Merge pull request #145 from natura-cosmeticos/update-la...	5 days ago
doc	docs: [DSY-000] Update typography doc with italic style de...	6 days ago
gradle/wrapper	chore(deps): update dependency gradle to v5.6.4	2 months ago
sample	feat: [DSY-000] - Refactor textfield label with 2 lines	4 days ago
tools	ci: [DSY-000] Upgrade lib version	4 days ago
.editorconfig	[DSY-701] Adding spacing tokens to themes	4 months ago
.gitattributes	Configuring Git LFS	9 months ago
.gitignore	Revert "Merge pull request #65 from natura-cosmeticos/fea...	4 months ago
.typo-ci.yml	chore: replace committed by committed on Typo CI	27 days ago
Makefile	ci: [DSY-1262] Remove script because snapshot is being...	16 days ago

About
Natura Design System - Nativo Android
Readme

Releases 21
Version 2.4.0 - Button... Latest
4 days ago
+ 20 releases

Packages 1
com.natura.android.designsystem 2.4.0

Contributors 15

github.com/natura-cosmeticos/natds-android