

Webから見た Alexa Presentation Language

音声UIにGUIを追加する

株式会社ZOZOテクノロジーズ
開発部 R&D担当
武田修平

Copyright © ZOZO Technologies, Inc.



ZOZO
Technologies

株式会社ZOZOテクノロジーズ
開発部
R&D担当



武田 修平 / sesta / せすた

- AlexaやGoogle Assistantなど音声UIの開発を担当
- 昔はWebサービスのフロントをやっていたが
最近はAlexaとずっとおしゃべりしている
- 趣味は庭いじりで、今年はメロンに手を出している

Amazon Alexa

Alexa が使える端末たち



Echo Dots



Echo Spot



Echo Show

Alexa Presentation Language

Alexa Presentation Language (以下、APL) とは

- Alexa用のデザイン言語
- 声でのやりとりを補強するビジュアルな表現が可能となる
 - 読んでいる内容を文章で並べる
 - コーディネートを画像で一覧表示
 - チュートリアルを動画で再生
- タッチでの操作を可能にしたり、間を置いてからしゃべらせるなど
インタラクションの幅が広がる

APLをふんだんに使ったスキルの紹介（宣伝）

着たいコーデが、話して見つかる。

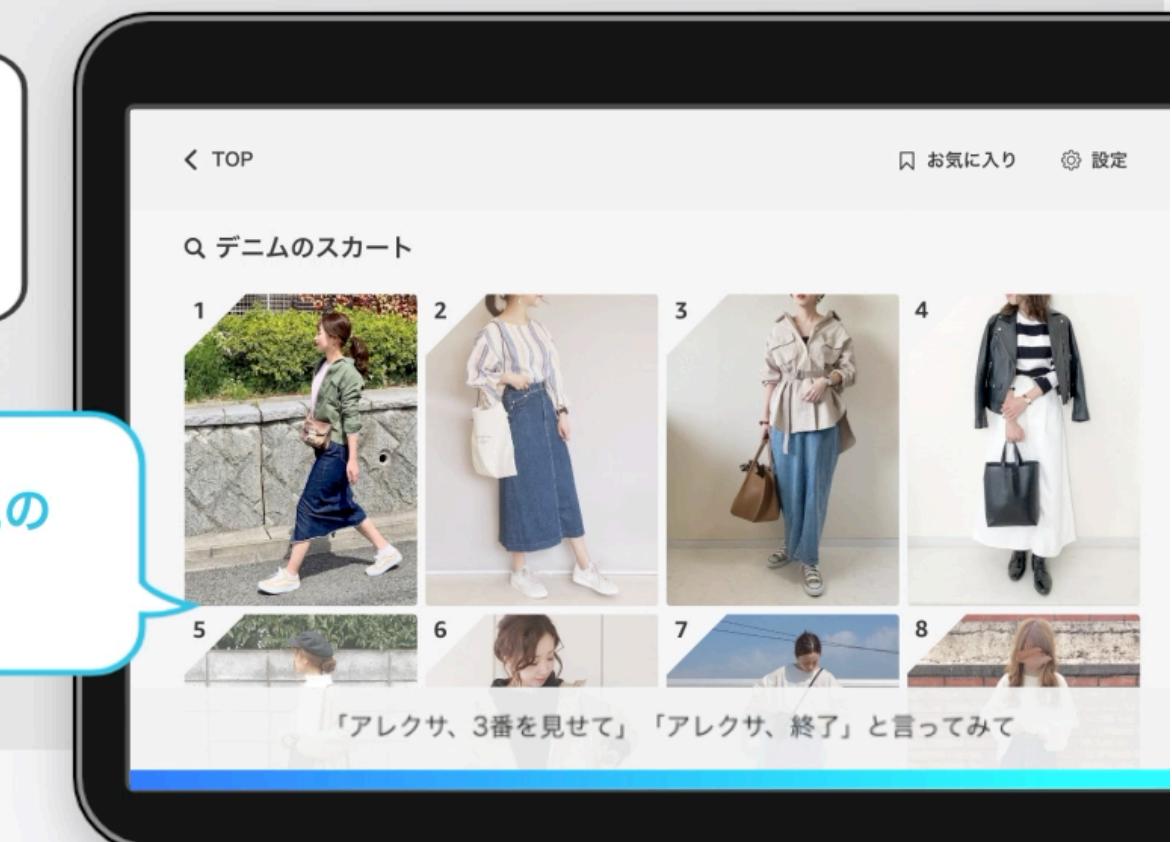


コーデ相談
by WEAR

アレクサに話しかけるだけで、着たいコーデが見つかります。

アレクサ、デニムの
スカートのコーデを教えて

はい！WEARで人気の
コーデを表示します



<https://www.youtube.com/watch?v=LaKYaX7Ci0A>

話すこと

1. APLを使ったレイアウトの組み方

- HTML/CSS にあたる部分

2. APLで可能な動的な表現

- JavaScript にあたる部分

3. APLでパーツの管理はできるのか

- Vue や React におけるコンポーネントのような仕組みの話

1. APLを使ったレイアウトの組み方

APLはオブジェクトの構造で記述するHTMLのイメージ

- 位置の計算方法はHTMLとほとんど一緒
 - 基本的に左上からスタックされていく
 - absolute もある
 - flex もある
 - Flexbox/Yogaをサポート: <https://yogalayout.com>
- 指定できるプロパティもHTML/CSSと似てる
 - プロパティの例: height / padding / justifyContent

レイアウト例: 画像を中央に表示

type で
コンポーネントの種類を指定
(≈ HTML のタグ)

見た目を整える情報を
それぞれ指定

```
{  
  type: 'Container',  
  
  item: {  
    type: 'Image',  
  
    source: 'https://image-url',  
  
    height: '100px',  
    width: '100px'  
  },  
  
  alignItems: 'center',  
  justifyContent: 'center',  
  height: '100vh',  
  width: '100vw'  
}
```

コンポーネント特有の
情報を指定

APLにおけるコンポーネントとは

- Alexaが標準で解釈できる、それ以上分解できない要素でWebにおけるHTMLのタグに近い存在
- コンポーネントごとに役割がはっきり決まっており、指定できるプロパティも異なる

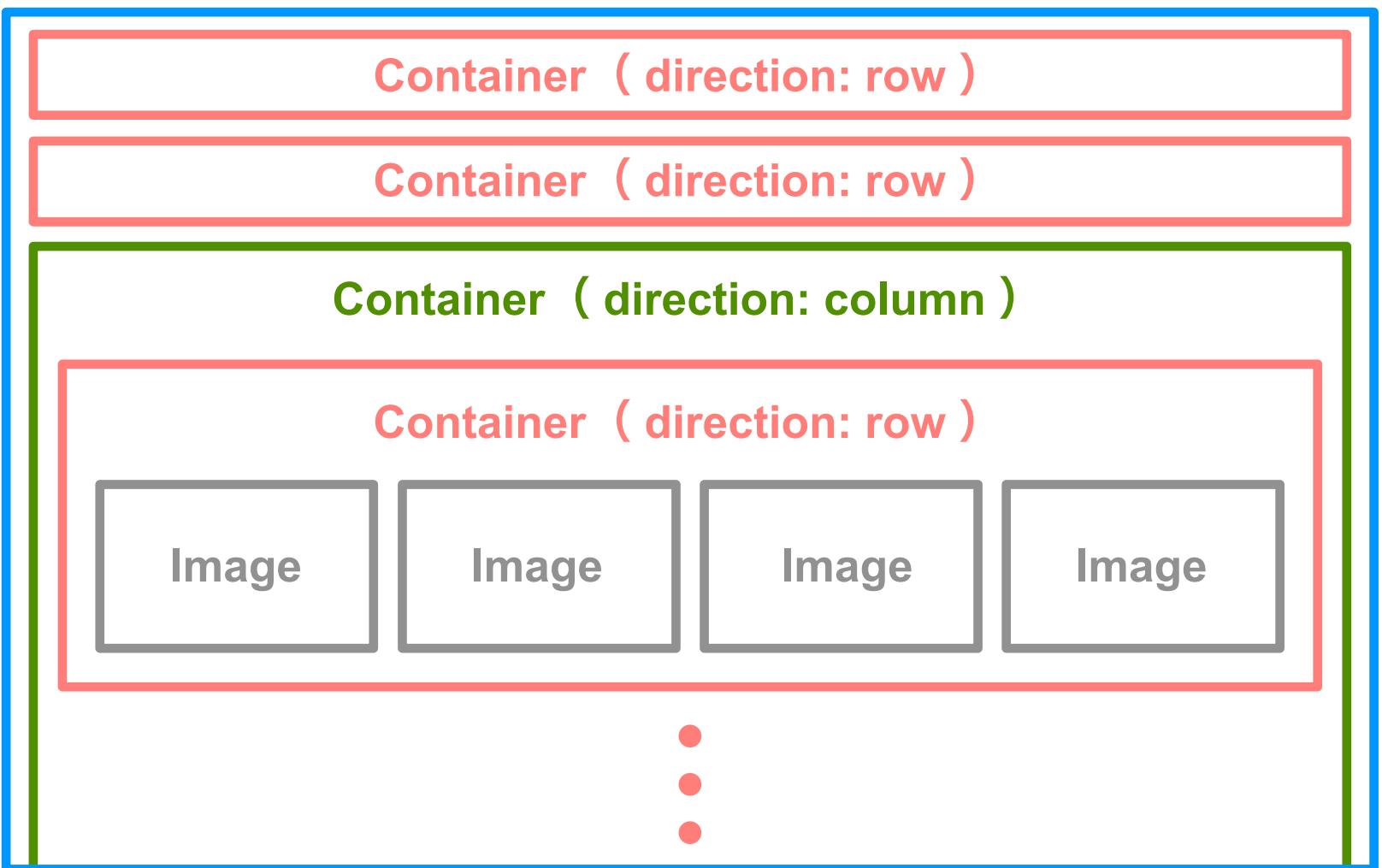
コンポーネントの例

- Container コンポーネント
 - Flexbox を用いたレイアウトで用いる
 - item というプロパティに渡すことで、他のコンポーネントを内包できる
- Text コンポーネント / Image コンポーネント
 - テキストや画像を表示
- ScrollView コンポーネント
 - 内包したコンポーネントが画面に収まらなかった場合にスクロールできるようにする

コーデ相談スキルの検索結果の構造



ScrollView



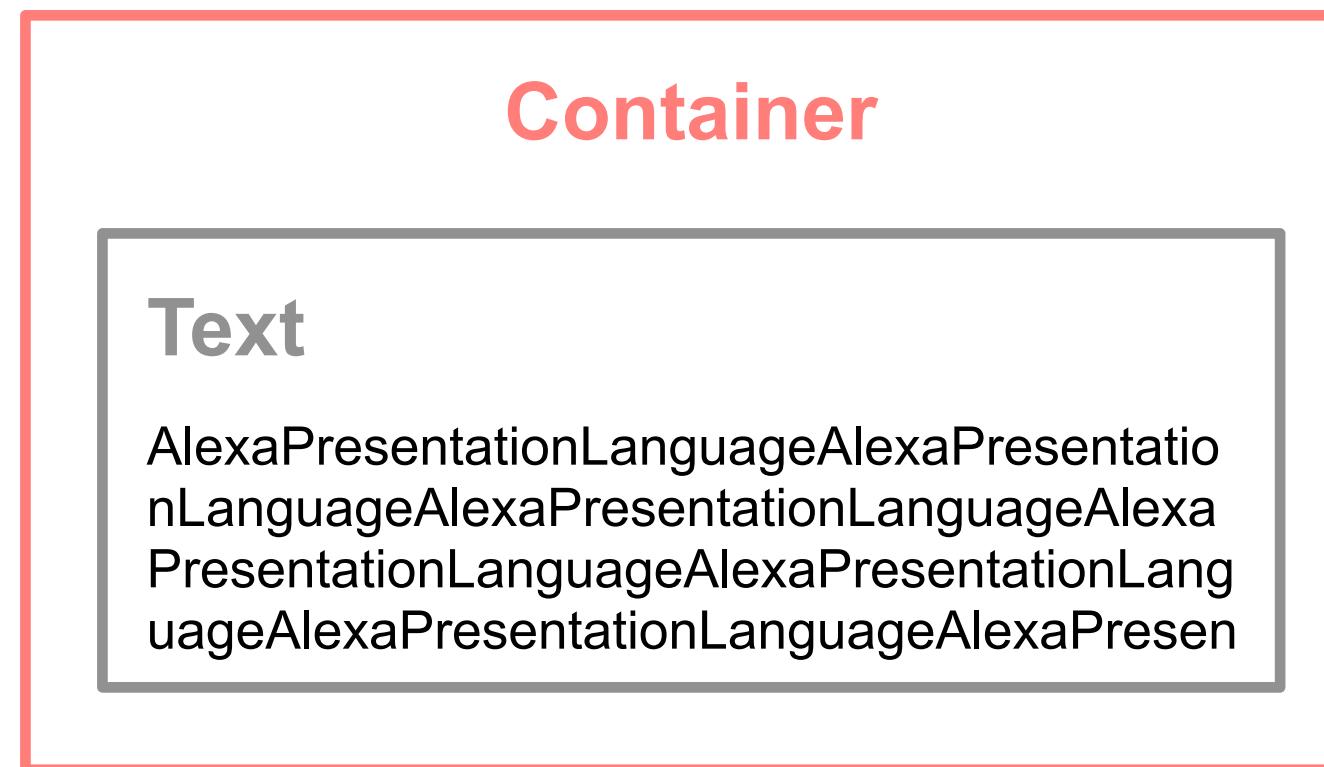
Webとの違い

- コンポーネント共有で使用できるプロパティが少ない
 - 見た目に関係するものだと width や height、padding だけが共通
 - 要素を並べる => Container、背景色やボーダーなどの装飾 => Frame
- リッチなコンポーネントが用意されてる
 - Pager: スワイプで中身を切り替えられるコンポーネント
 - Sequence: 遅延読み込みを自動でやってくれるコンポーネント

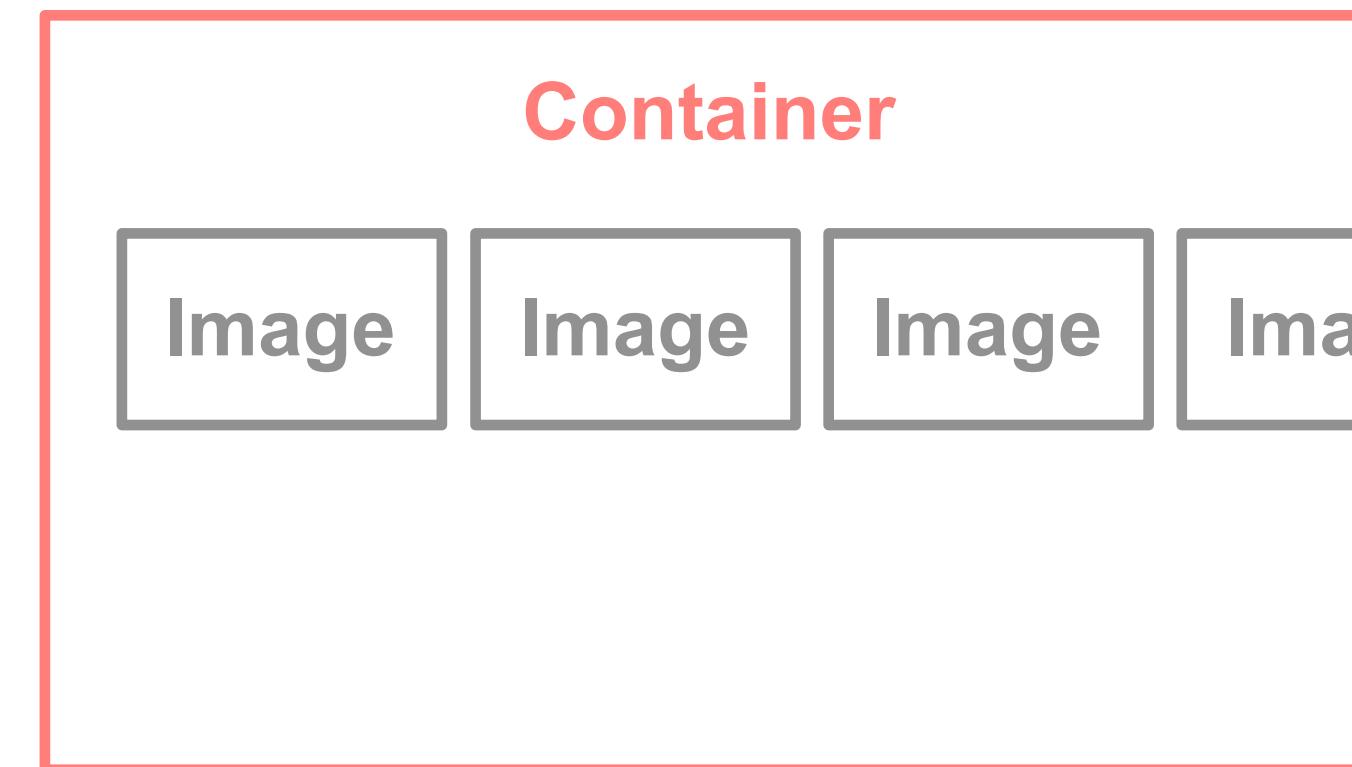
Webとの違い

- ・折り返し (wrap) が弱い
 - ・テキストは改行してくれるが、コンポーネント自体がはみ出たら overflow: hidden;
 - ・デバイスごとにコンポーネントを切り替える仕組みを使って気合でがんばる

テキストの場合



コンポーネントの場合



2. APLで可能な動的な表現

今回触れる 動的な表現たち

- ・タッチなどインタラクション周り
- ・アニメーション

タッチなどインタラクション周り

- ・コンポーネントがデフォルトで操作可能にしてくれているもの
 - ・スワイプ
 - Pager コンポーネントによるコンテンツの切り替え
 - ・スクロール
 - ScrollView コンポーネントなど
- ・開発者が自分で動きを指定するもの
 - ・タッチ
 - TouchWrapper コンポーネントでタッチイベントを監視する
 - ・スワイプ / スクロール (ドキュメントに載ってないけど動かなくはない)
 - ・onPageChange / onScroll プロパティ 詳細: <https://techblog.zozo.com/entry/alexapresentation-language>



APLは描画以外もできる

- ・コンポーネント
=> 描画の内容を決めるもの
- ・コマンド
=> 動きを与えるもの

コンポーネント

コマンド

```
{  
  type: 'ScrollView',  
  
  item: {  
    type: 'Container',  
  
    height: '1000vh'  
  },  
  onScroll: [  
    {  
      type: 'SetValue',  
  
      componentId: 'image1',  
      property: 'opacity',  
      value: '0.4'  
    },  
  ],  
}
```

APLにおけるコマンドとは

- ・画面上に描画されたコンポーネントを変更したり
音声をコントロールする命令
- ・画面を描画する時や
インターラクションに合わせて実行させることができる

コマンドの例

- Scroll コマンド
 - ScrollView などスクロール可能なコンポーネントで任意の位置までスクロールさせる
- SpeakItem コマンド
 - 指定した文章をAlexaに発話させる
- Parallel コマンド
 - 複数のコマンドを並列で実行させる

アニメーション

- ・工夫すると以下のことはできる
 - ・順番に表示
 - ・透明度を連続的に切り替え
 - ・下から生やす（トースト）



気をつけなくてはいけない点

- プロパティはほとんど書き換えられない
 - SetValue というコマンドを使えばプロパティの書き換えは可能
ただし opacity と text のみ（2019年6月時点）
- 値を連続的に変化させられない
 - SetValue で opacity を指定しても、一瞬でその値になってしまう
CSS の transition のようなプロパティは無い

極めたい方へ

- 「Alexa Presentation Languageの限界を超えて、アニメーションや相槌を実現する」
 - <https://techblog.zozo.com/entry/alexapresentationlanguage>



3. APLでパーツの管理はできるのか

回答：仕組みがちゃんとある

パートの切り分けと再利用

- APLレイアウト という仕組みを使う
 - <https://developer.amazon.com/ja/docs/alexa-presentation-language/apl-layout.html>
- APLレイアウトは複数のコンポーネントから成る固まりに対して任意の名前を付与できる仕組み
 - 作ったレイアウトは、type: "MyContainer" のように普通のコンポーネントのように呼び出せる
- Vueなどのようにプロパティとして引数も渡せる

見た目に関するプロパティの切り分け

- Styles という仕組みを使う
 - <https://developer.amazon.com/ja/docs/alexa-presentation-language/apl-style-definition-and-evaluation.html>
- color や fontSize など複数のプロパティを1つのスタイルとして任意の名前をつけることができる仕組み
 - 例えば AccentStyle というスタイルを用意して、文字色とサイズを指定しておけばコンポーネントで style: "AccentStyle" のように指定するだけでそのデザインを利用できるようになる
- 複数のスタイルは指定できないが、スタイルの継承ができる

管理できそう？

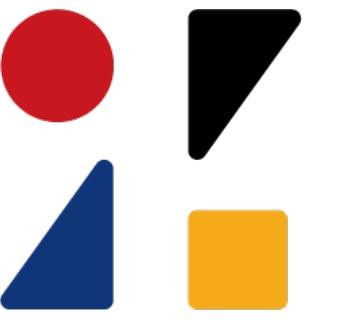
微妙に手が届かない現実

- APL レイアウト
 - コマンドをレイアウトの中に留めておけないので参照が増えやすい
 - レイアウト外のコンポーネントのタッチ時や描画後などに実行したいコマンドがあったらどうしようもない
- Styles
 - height や padding、alignItems などは指定できない
 - 文字のスタイルの設定ぐらいならいけるが、ボタンは厳しい
 - 値を変数化する仕組みはあるので
レイアウトと組み合わせれば いけなくはない

まとめ

- 静的な画面なら割と再現が可能
 - ・アニメーションなどの動的な要素も工夫すればいける
- コンポーネントを管理できる仕組みもあるので
シンプルなスキルならOK
- HTML5に対応予定なので、それまで無理しないのが安全

Let's Voice UI



ZOZO
Technologies

ZOZOTOWN

<https://zozo.jp/>



- ・日本最大級のファッショントショッピングサイト / アプリ
- ・1,200以上のショップ、7,000以上のブランドの取り扱い
(2019年3月末時点)
- ・常時73万点以上の商品アイテム数と
毎日平均3,200点以上の新着商品を掲載
- ・即日配送サービス
- ・ギフトラッピングサービス
- ・ツケ払いなど



WEAR

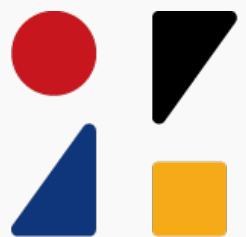
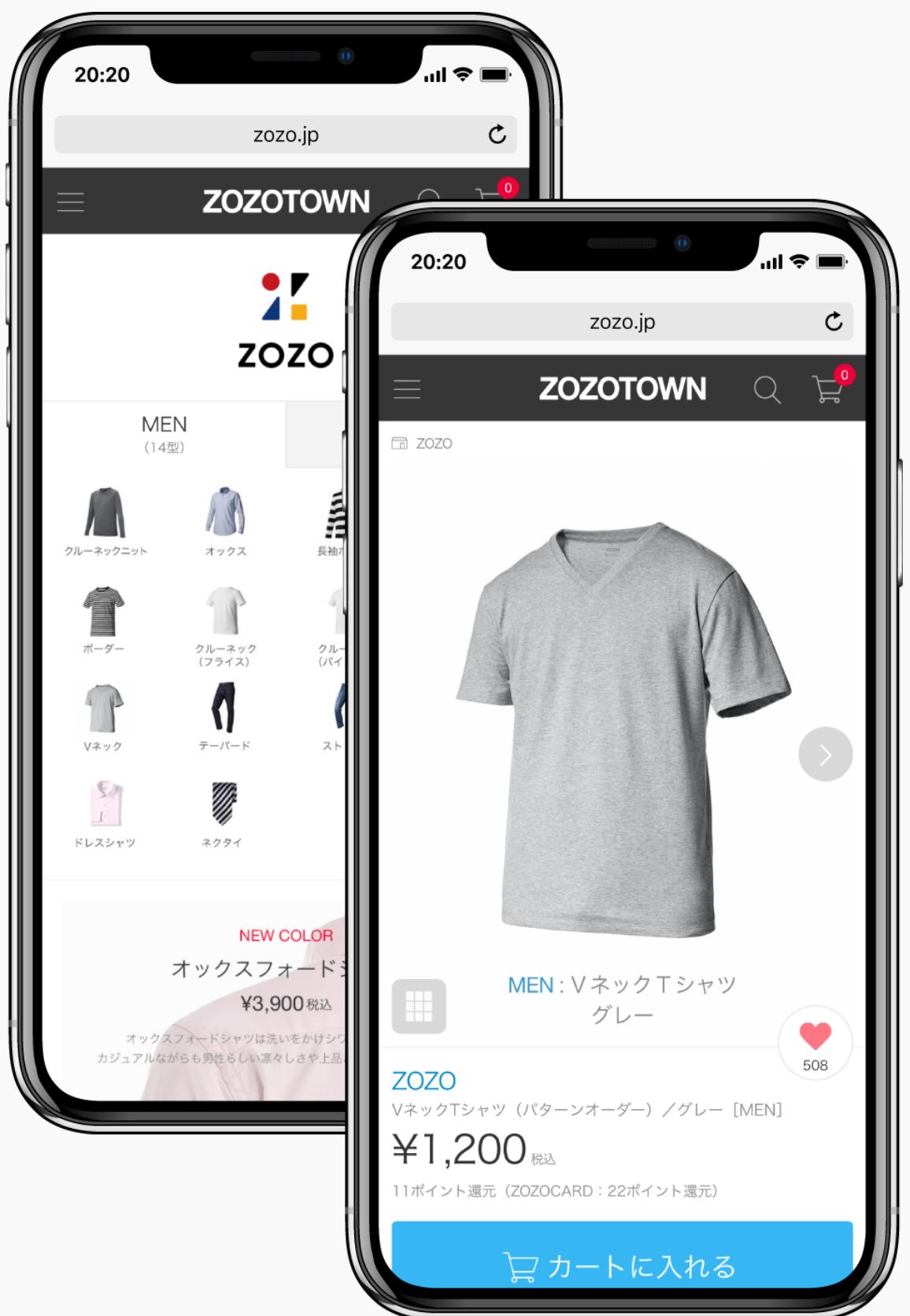
<https://wear.jp/>



- ・日本最大級のファッショナブルコーディネートアプリ
- ・1,300万ダウンロード突破
コーディネート投稿総数は800万件以上
(ともに2019年3月末時点)
- ・全世界 (App Store / Google Playが利用可能な全ての国) で
ダウンロードが可能
- ・10万人以上のフォロワーを持つユーザー (WEARISTA) も誕生

 Download on the
App Store

 GET IT ON
Google Play



ZOZO

<https://zozo.jp/brand/zozo/?dord=21>

- ・ 「ZOZOSUIT」で計測した体型データをもとに、一人ひとりの体型に合った「あなたサイズ」のアイテム
- ・ 「究極のフィット感」を実現したベーシックアイテムを提供
- ・ アイテム：Tシャツ / デニムパンツ / シャツ / ビジネススーツ / ネクタイ / ボーダーTシャツ / 長袖クルーネックTシャツなど