

マテリアル デザインの発表とリリースから 2 年以上が経過しました。この間、多くのパートナー様がマテリアル デザインのデザイン パターンとガイドラインを導入して成功を収めています。コンテンツに奥行き概念が導入されたことから、マテリアル デザインへの移行は複数の段階に分けて行われることが多く、「アプリを更新する際にはどのガイドラインを優先すべきか」という質問が多く寄せられています。このようなご質問にお答えするため、アプリにマテリアル デザインを導入する際の第一段階での手順をご紹介します。

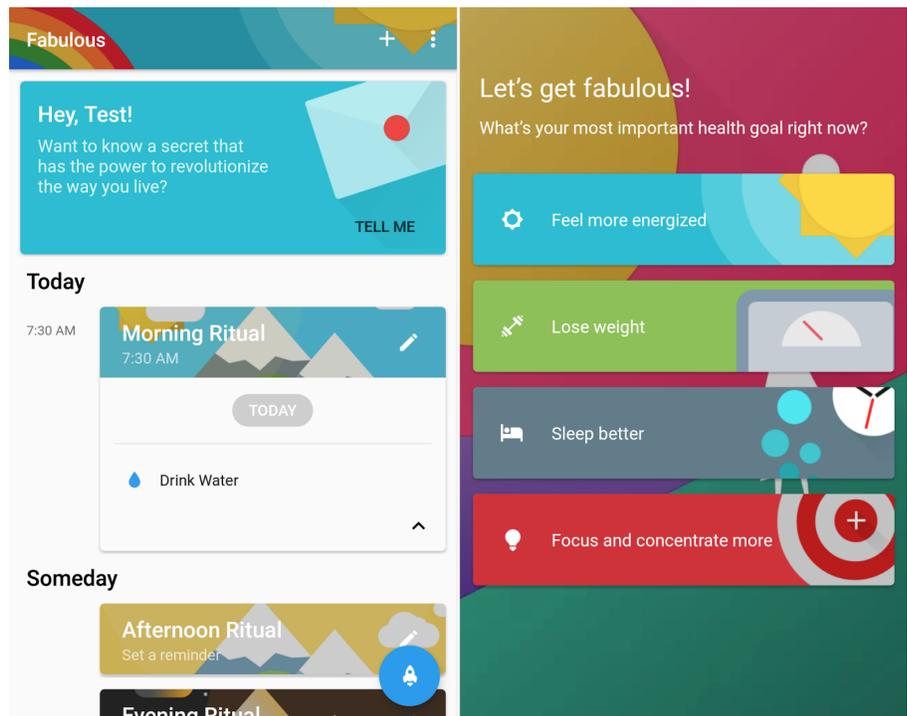
マテリアル デザイン導入のステップ

サーフェスと影

マテリアル デザイン対応アプリの UI 要素は、階層化されたサーフェスが、縦方向(Y軸)または横方向(X軸)にそれぞれ重ねられている必要があります。影を使ってサーフェスの奥行き(Z軸)を表現することで、視点をコントロールし、階層関係を表現することができます。[詳しくはUIの奥行きと階層に関する詳細をご覧ください。](#)

手順 1

アプリを更新する際には、次の基本原理が UI やコンテンツに適用されることを念頭に置いてください。



[Fabulous](#) では影と目立つ表示面を使って、ユーザーの視点を集めています。

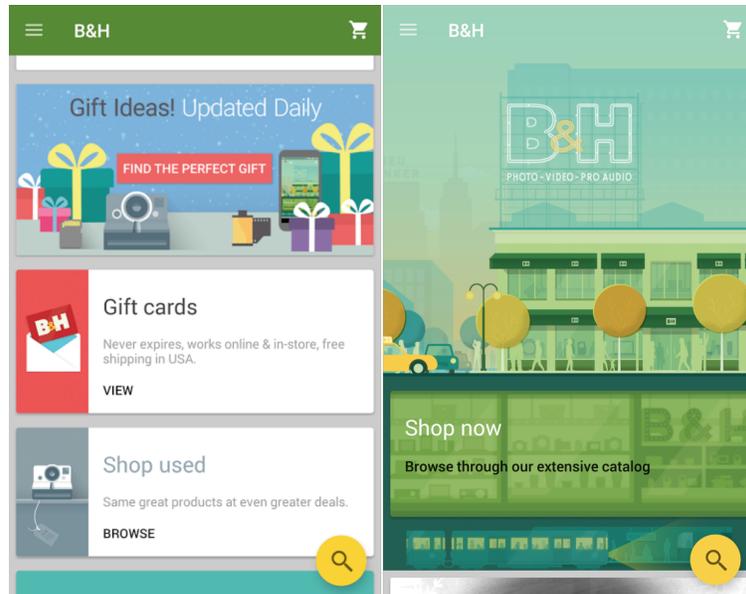
作成日: 06/01/2016

バージョン: 0.1

※本資料の情報は作成日時点のものであり、最新の Android や Google Play の仕様やガイドラインとは異なる場合がありますのでご了承ください。

画像

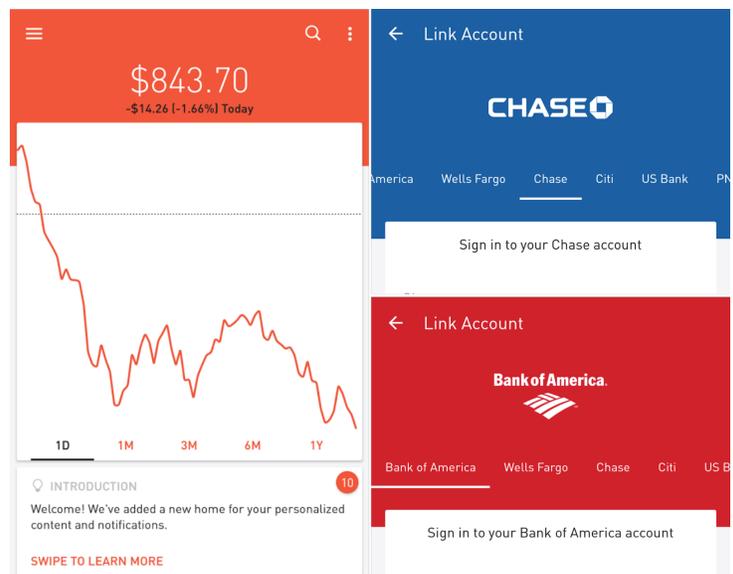
マテリアルデザイン対応アプリの画像は、余白なしにする事を推奨しています。UIの没入感を高めスクリーン内の情報を見やすくするために、アプリでは、写真の間のマージンや、写真と画面の端との間の余白を最小限に抑える必要があります。



[B&H](#) では、フルブリード画像でユーザーの目を引く没入感を表現しています。

色

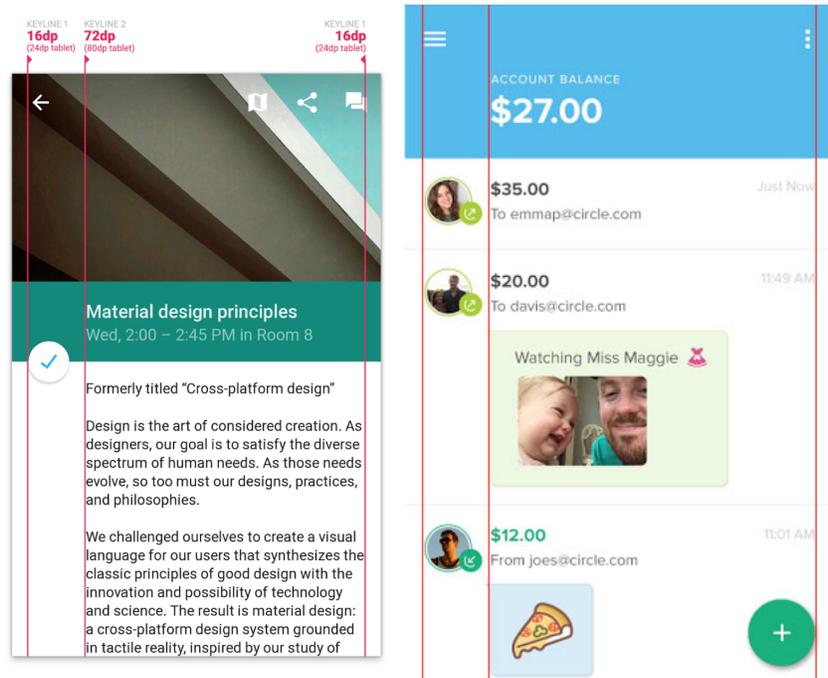
ブランドを強調するためにも、Toolbar や重要な UI 要素に色を大胆に使いましょう。



[Robinhood](#) では統一感のある色を使うことで、ブランドや UI 要素をユーザーにアピールしています。

指標とキーライン

読みやすさと視覚的な一貫性を持たせるため、UI 要素の配置とサイズ設定には 8 dp の [ベースライングリッド](#) を使用する必要があります。テキストをより正確に配置する場合は、4 dp のグリッドを使用できます。必要に応じて、本文テキスト、サムネイル、アプリバー タイトルなどの要素を [3つのキーライン](#) に揃えます。



[Circle Pay](#) では、ユーザーが主要な情報をすぐに見つけられるよう、コンテンツをキーラインに合わせて適切に配置しています。

適切なアニメーションとインタラクション

アニメーションと遷移を使用する事でアプリ内でユーザーをガイドする事ができます。適切なアニメーションは心地よいユーザー エクスペリエンスを提供し、視覚的なつながりを強めるうえでとても有効です。Android 5.0 (API レベル 21 以降) で提供される [カスタムアニメーション](#) をご活用ください。またアニメーションの振る舞いに関しては [マテリアル モーション](#) もご参照ください。

マテリアル デザイン対応アプリでの基本的なタップのフィードバックは、[表示面のリップル \(波紋\)](#) 形式とする必要があります。5.0 以前の端末ではリップル機能を使用できないため、デフォルトの android.selectableItemBackground の動作となります。

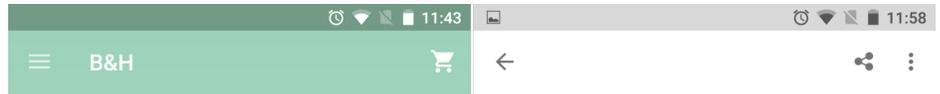
- 表示面のリップルの他に、マテリアル デザインでのインタラクティブな表示面は、入力中にユーザーの指に合わせて [タップ時に隆起](#) させることもできます。

手順2

マテリアル デザインの主な基本原理を考慮する際、アプリ更新の実際の手順としては、まず、選択するシステム UI コンポーネントが、新しい視覚的デザインのガイドラインと操作パターンに準拠しているかを確認します。

[システム ステータスバー](#)

Android 5.0 以降では、ステータスバーの色に、アプリのメインカラーや現在の画面のコンテンツを目立たせるような色を使用する必要があります。フルブリード画像の場合は、ステータスバーを透明にすることもできます。詳しくは、こちらの [Stack Overflow](#) の投稿をご覧ください。API レベル 23 以降では、システム ステータスバーに[明るい色](#)を使用できます（アイコンは暗色になります）。



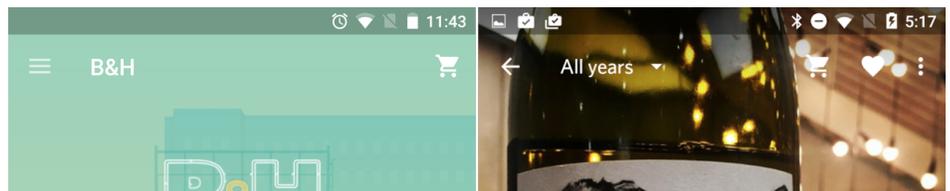
システム ステータスバーの 2 つの例: 左の例ではアプリバーを目立たせる色、右の例では明るい色のステータスバーが使用されています。

[アプリバーとツールバー](#)

以前のバージョンの Android のアプリバーとは異なり、マテリアル デザインのアプリバーでは左上にランチャー アイコンを表示しません。代わりに、色、タイポグラフィ、プロダクトロゴを使ってブランディングします。



大きな画像が表示される画面では、画像が画面の端から端まで表示され、また[アプリバーやステータスバーの背後に表示される場合もある](#)ことから、アプリバーを透明にして UI の没入度を高める必要があります。透明なアプリバーのアイコンでは、必ずテキストを保護してください。新しい [Toolbar](#) ウィジェットは透明にして、レイアウト内に直接配置できます。

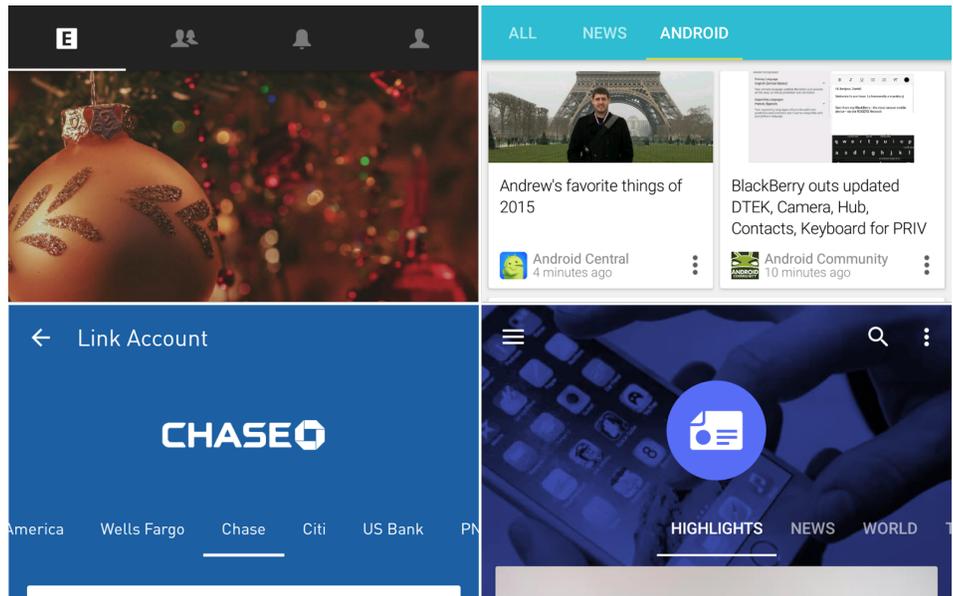


[B&H](#) と [Delectable](#) のアプリでは、透明なアプリバーとともに大きな画像を表示しています。

タブ

タブは、マテリアルデザインの[操作とスタイル](#)に準拠している必要があります。タブは横方向にスワイプできるようにします。タブ間に縦の区切りを設けることはできません。選択状態のタブは、アクセントカラーを使って、前面の色を変更したりタブテキスト（またはアイコン）の下に小さい線を表示したりすることで、[選択されていることを示す](#)必要があります。

階層内で明確に区別されていない限り、通常、タブとアプリバーの表示面の高度は同じです。



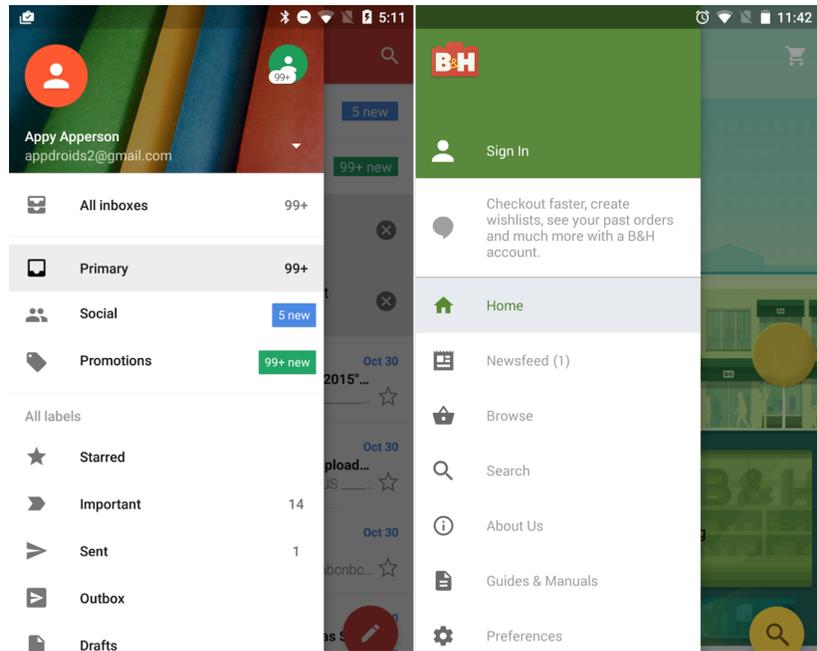
このように、タブには用途やコンテキストに応じてさまざまな表示方法があります（[EyeEm](#)、[Palabre](#)、[Robinhood](#)、[Google Play ニューススタンド](#)）。

ナビゲーションドロワー

ナビゲーションドロワーには最上位のナビゲーション要素のみを含める必要があります。また、新しい[視覚的デザインのガイドライン](#)に準拠している必要があります。

ナビゲーションドロワーは、アプリバーと画面コンテンツよりも前で、システムステータスバーの後ろに表示する必要があります。実装について詳しくは、[こちらの画像の例](#)と、こちらの [Stack Overflow](#) の投稿をご覧ください。

機能を見つけやすくするため、アプリの初回起動時はドロワーを開くようにすることをおすすめします。



[Gmail](#) と [B&H](#) による、マテリアル デザイン ナビゲーション ドロワーの適切な実装例

手順3

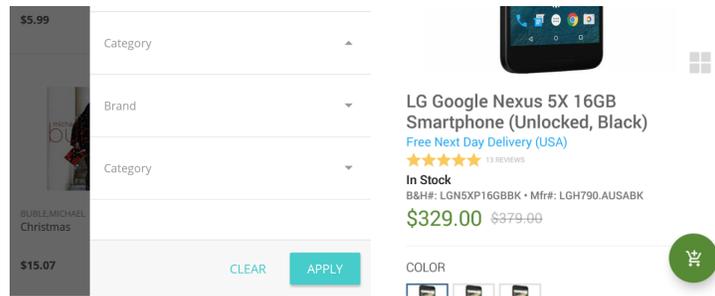
システム UI コンポーネントを更新したら、次は UI 要素を更新します。アプリの更新にあたって、ボタンのスタイルやコンテンツのレイアウトから UI のテーマまで、いくつかの留意点があります。

ボタン: ボタンはアプリ内で最もよく使われるインタラクティブ要素であるため、ボタンを最新の視覚的デザイン パターンに合わせて更新することは、最もインパクトの強い変更の 1 つとなります。

カスタムボタンを使用する場合は、角を丸くし過ぎること、過度のグラデーションを使用すること、光沢のある表示面を使用することは避けます。一般に、こうしたタイプのボタンを使用するとアプリが時代遅れに見えます。

マテリアル デザインのボタンは、重要度や視覚的なスタイルに基づき次の 3 つのカテゴリに分かれます。

- [Floating Action Button](#) は、各画面での最も重要な操作に用いるボタンで、視覚的に明確に認識できます。
- [Raised Button](#) は、視覚的な強調が必要な重要な操作や、テキスト量の多いレイアウトでフラットボタンだと目立たないときに用いるボタンです。
- [Flat Button](#) は、アプリ内で最も一般的に用いられるボタンで、強調する必要のない簡単な操作に使用します。



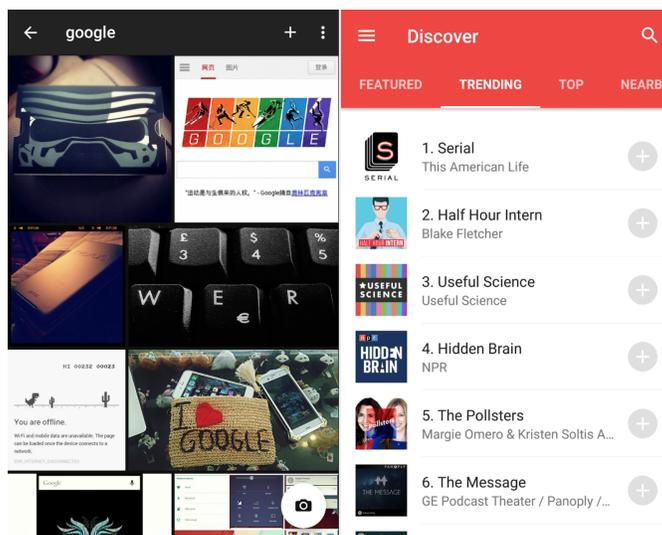
[Jet](#) と [B&H](#) では、主要な操作の強調と、強調の解除にさまざまなタイプのボタンを使っています。

アプリでは、必要に応じて円形の Floating Action Button (FAB) を使って、画面の主要な操作を強調する必要があります。FAB は、他のすべての UI 要素の上にフローティング表示され、視覚的に目立つボタンです。FAB は、その画面内に主要な操作がある場合、または (FAB で展開できる) 複数の関連する主要な操作がある場合にのみ使用してください。また、アプリの機能が FAB を必要としない場合は、必ずしも FAB を使用する必要はありません。

カード、タイル、リストの比較: コンテンツの表現は、ユーザー エクスペリエンスを左右する重要な要素です。アプリの更新にあたって、適切なコンテンツ表示方法を選択することで、適切な対象に注目を集めることができます。

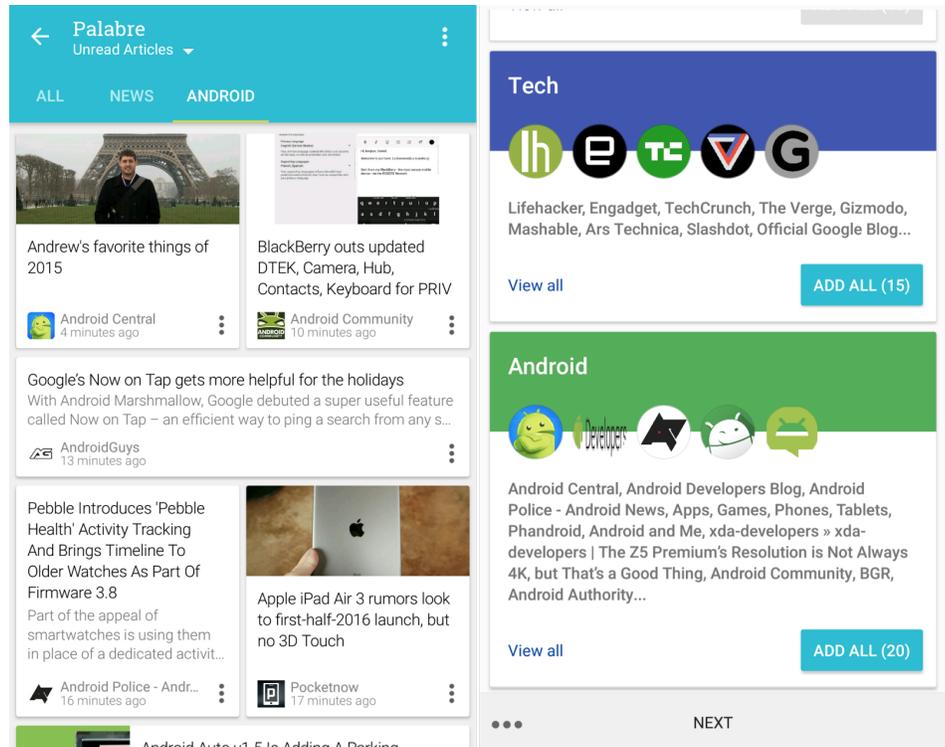
ユーザーが複数の項目を直接比較する場合には、リストやタイルを使用します。

- リストビューは、テキストの量が多く、すばやく流し読みできるコンテンツに使用します。たとえば、ごく小さな画像を伴った記事の一覧などに使用できます。
- タイルは、ユーザーが閲覧する主要な視覚的コンテンツがある場合に使用します。



[EyeEm](#) と [Pocket Casts](#) ではシンプルなリストビューを使って、コンテンツの概要を把握しやすい形式でユーザーに表示しています。

マテリアル デザインのカードは、コンテンツの種類が混在している場合（ユーザーが画像やテキスト文字列を直接比較せず、コンテンツを探すような場所）に使用します。また、表示面で複数の操作が行える場合にも使用します。



[Palabre](#) では、さまざまな種類のカードを使って、多彩なコンテンツを表示しています。

コンテンツのビューでは「カードを使いすぎない」ようにしてください。たとえば、シンプルな一連のテキスト記事の表示にはカードを使用しないでください。カードを使用すると、項目を目立たせて注目を集められますが、カードを使いすぎると見にくくなるので逆効果です。

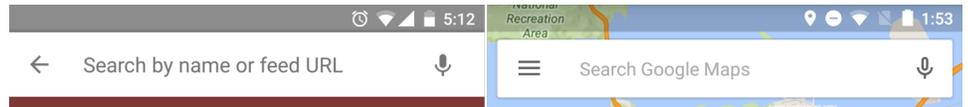
テーマ: アプリがOSのUIと視覚的に協調するように、デフォルトのシステムテーマを[最新のマテリアルテーマ](#)に更新する必要があります。システムテーマを更新すると、洗練された外観へと手軽に変更できます。

一貫した視覚的デザインとユーザー エクスペリエンスを提供するため、Gingerbread および Holo のテーマは、必ず最新のテーマに更新してください。

[セレクトタ](#)、[ピッカー](#)、[ダイアログ](#)、[メニュー](#)、[テキスト欄](#)などの UI 要素のテーマを、マテリアル デザインの標準に更新する必要があります。

検索

マテリアル デザインのアプリ内検索は、アプリバーのインライン検索を使って実装する必要があります。アプリの主要な機能がコンテンツの検索である場合は、専用の検索ウィジェット アプリバーの使用を検討します。



シンプルなインライン検索バーは、アプリ内の検索時でもユーザーの関心が他に移動することがありません。一方、専用の検索アプリバーは、アプリのメイン機能が検索であることがすぐにわかります ([Pocket Casts](#)、[Google マップ](#))。

デバイダーの使用

デバイダーは、コンテンツの主要セクション同士を分ける場合や、テキスト領域同士の間重要な区別を設ける場合にのみ使用します。シンプルなりストビューで項目ごとにデバイダーを使用すると、通常、読みやすさは多少向上しますが、大部分が見にくくなるため注意が必要です。代わりに、空白スペースやタイポグラフィを使ってコンテンツを区別することをおすすめします。

スワイプによる更新

スワイプによる更新は標準の Android パターンであり、再読み込みの必要なコンテンツ画面がある場合に実装する必要があります。

カスタム アニメーション

共有要素の遷移と円形状の出現は、現在の画面と次の画面につながりを持たせる方法として適しています。

スクロールの手法

より快適な表示をユーザーに提供するため、さまざまなスクロール手法を試すことを検討してください。

本ガイドで紹介した内容は、アプリをマテリアル デザイン対応アプリとして更新する際の基本的な手順として、ぜひご活用ください。今後もマテリアル デザインの効果的な導入方法について、お知らせしていく予定です。

手順 4

アプリの更新時には、Android 5.0 とマテリアル デザインで導入された次のようなアップデートと新しい UI パターンについて考慮する必要があります。