

# Apps Design Drill

ver.1



## はじめに

このドリルは、クラスメソッドのアプリ制作におけるUIデザインの基本が学べるように設計されています。

UIデザイン未経験者の方が、このドリルを完了することで最低限の知識が得られる構成になっています。

各章の問題を解いて、解説を読むと理解できる参考書形式です。

ドリルをやりとげる小さな成功体験を糧に、UIデザインの基礎能力を鍛えていきましょう。

ここに書いてあることがすべてではありません。

想像力を働かせ、人とコミュニケーションを取りながらデザインすることを楽しみましょう。

## このドリルのターゲット

ターゲットは、体験をもとに画面と遷移を設計し制作する「UIデザイナー初心者」です。

※スマートフォン及びタブレットのUIデザインを対象としています。

## ドリルに書いていないこと

- デザインツールの細かい使い方
- カッコいいUI の作り方
- 人間中心設計 (HCD)
- UX手法
- WEB
- テレビ
- ウェアラブル
- 音声
- AR
- VR

# 目次

## PART 1 : UIデザイナーの役割

### PART 1 問題

- 1-1. クラスメソッドUIデザイナーのマインド
- 1-2. 基本の姿勢
- 1-3. デザイナーの時間管理

### PART 1 問題解答

## PART 2 : デザイナーの動き

### PART 2 問題

- 2-1. プロジェクトの受注
- 2-2. 要件ヒアリング
- 2-3. 開発手法によるスケジュール確認
- 2-4. 必要最低限の項目を洗い出す
- 2-5. ワイヤフレームで全体像を確認
- 2-6. メイン機能の動線（目的達成まで）を再度情報設計・デザイン

# 目次

2-7. 1機能ごとディテールを詰めて確認していく

[PART 2 問題解答](#)

[PART 3： デザイン前の前提知識](#)

[PART 3 問題](#)

3-1. 著作権、意匠権、ライセンス

3-2. 知っておくとよい知識

[PART 3 問題解答](#)

[PART 4： アプリデザイン基礎知識](#)

[PART 4 問題](#)

4-1. 最初はOSの違いを理解しましょう

4-2. アプリと解像度の理解

4-3. プラットホームに応じたデザイン知識

[PART 4 問題解答](#)

# PART 1 UIデザイナーの役割

チーム開発において、

UIデザイナーに求められる一番のマインドは「協働」です。



## この章の目的

1. デザイナーの役割を理解する
2. メンバーとコミュニケーションをする上での基本的な考え方を理解する
3. チームにおけるデザイナーの時間を理解し、想像力をもってスケジュールを考えられるようになる

## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問1 UIデザイナーのチームメンバーになる可能性がある人をすべて選択してください。

- ア. クライアント（依頼者）
- イ. プロジェクトマネージャー
- ウ. プロダクトオーナー
- オ. スクラムマスター
- カ. サーバーサイドエンジニア
- キ. フロントエンドエンジニア
- ク. UXデザイナー
- ケ. インフォメーションアーキテクト



## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問2 以下の文章を読み、正しいものをすべて選択してください。

- ア. デザイナーの仕事は、自分を表現することである。
- イ. デザイナーの仕事は、ユーザーがそのサービスを使用するときに、無理せず空気のように使用できるにはどうしたらいいかを考えることである。
- ウ. デザイナーの仕事は、見た目をの美しさのみを追求する仕事である。

まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問3 木曜日のMTGでクライアントに4日後にデザインを提出してくださいと言われた場合、デザイナーとしての確認すべき項目をすべて書き出してください。（記述問題）

## PART 1：UIデザイナーの役割

デザイナーにも、種類と役割があります。

### UXデザイナー

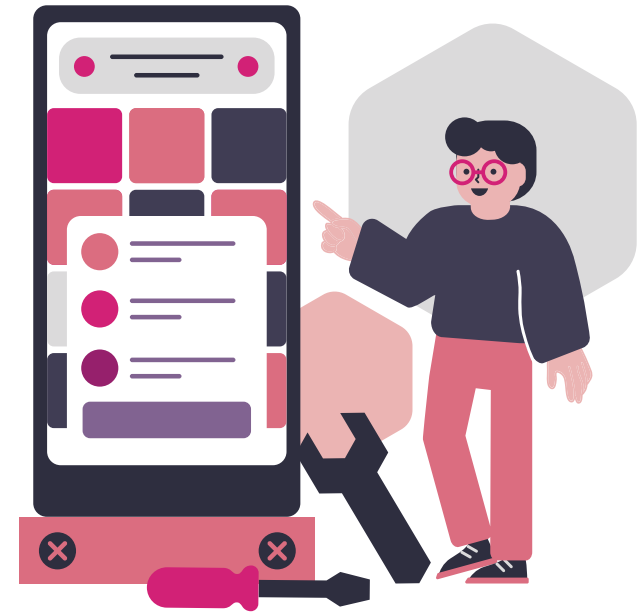
ユーザーの体験設計を行うデザイナーです。ユーザー行動に基づいてより良い体験を届けるための設計をします。UIデザイナーと密接に関係し、一緒に考えてUIを組み立てる人です。

### UIデザイナー

プロダクトの目的や意図を理解し、各種ガイドラインにそった、最適で実現可能なデザインを組み立てる人です。先述していますが、このドリルのメインターゲットです。

画面を作成するだけでなく、ユーザーの動線を考えたり、デザインシステムなど協働のためのデザインを行うデザイナーのことです。

そのほかに、情報設計（IA）であったり、ビジュアルをメインに活躍するグラフィックデザイナー、出版物などを構成するエディトリアルデザイナーなど、デザイナーにも種類と得意・不得意などの分野が存在します。



## 1-1. クラスメソッドUIデザイナーのマインド

アプリ開発において重要なのは、チームであることです。

- プロジェクトマネージャー（PM）やエンジニアだけでなく、クライアント（お客様、社内開発の場合は社内のステークホルダー）も含めた全員の協働チームである意識をもちます。
- 商業デザイナーは、アーティストとは違い自分のためではなくユーザーのためにデザインを作成します。エンドユーザーの味方であることを忘れずにデザインを行いましょう。

### 1-1-1. WEBデザイナーとアプリUIデザイナーの差異

- WEB開発のようにディレクターがいて指示を出してくれたりすることはありません。代わりにPMやプロジェクトリーダーなどが存在しますが、対等な立場で対話をします。UIデザインは、アプリデザイナーが主導します。ディレクションも自分が兼ねていると考えて動きましょ。
- WEBはメインページのテンプレートを作成することで、量産体制をとることが可能ですが、アプリは機能ごとの最適UIを設定していくため、コンポーネント化は可能ですが、基本は全てのページを作成します。

### 1-1-2. プロジェクトで関わる人たち

- プロジェクトマネージャー（PM）
- プロダクトマネージャー（PdM）
- プロジェクトリーダー（PL）
- フロントエンドエンジニア
- サーバーサイドエンジニア
- スクラムマスター
- クライアント
  - PO（プロダクトオーナー）
  - 企業のブランディング担当者など

## 1-2. 基本の姿勢

チームメンバーとの関わり方の基本姿勢を記載していきます。

- メンバーに上下はなく、対等な立場でプロのデザイナーとしての意見を求められます。
- わからない用語・事柄は素直に聞きましょう。
  - 聞かなかった場合、理解できるものだと認識されますので、後から自分が困ります。
- 五月雨ではなく聞きたいことをまとめてから、ミーティング（MTG）をセットします。
  - 相手の時間を奪わないように心がけます。クライアントは特に、アプリ以外の仕事も兼任されているかもしれません。

- エンジニアはデザイナーと同じく無から有を生み出すクリエイターなので、集中する時間を大切にしています。

特にエンジニアとのコミュニケーションは重要です。

- デザイナーが思いつかない部分を担ってくれるパートナーである認識で接しましょう。
- 考えたデザインの実装が難しい場合、代替案を聞いてみましょう。もしくは一緒に考えてほしい旨を伝えてみましょう。

## 1-3. デザイナーの時間管理

デザイン後のタスクにメンバーが関係することを意識します。

デザイン作業で時間を使ってしまうと、締切日程を動かさない場合にはエンジニアが対応する実装時間が削減されます。UI微調整の時間や、エラー画面など対応が入ってくるため、その分スケジュールが遅延してしまう可能性が高くなります。

クライアントへ確認を行う前にエンジニアと実装可能かを確認しましょう。

クライアント確認後に実装不可とわかった場合、クライアントの時間を奪ってしまうこととなり、信頼度低下を招いた入り、スケジュール遅延の要因になります。

実現確認のことを「フィジビリティチェック」といいます。



## 1-3. デザイナーの時間管理

クライアントとのデザイン合意は全体の流れで合意をとってから詳細デザインFIX

エンジニアとの合意後、クライアントとの合意をとります。

エラー画面を除いた正常系メイン機能の動線（流れ）とデザイン方針を決定してクライアントと合意し、各画面のデザインFIXを目指します。（※FIX：デザイン業界で「調整完了」の意味で使用される用語）

はじめから1画面ごとに近視的なデザイン修正の対応を行うと『A画面で線をつけるデザインを行ったが、B画面では線がない』などの「全体のデザインの不整合（デザイン負債）」が発生します。共通デザインが使用できないため、実装工数も上乘せされていきます。

また、見た目についてはクライアントも指摘を行いやすく、このルールでデザイン運用しているといった論理的な回答ができない場合、統一性のないデザインが完成する原因になります。

そのほか、全体でデザインを捉えていない場合、チームメンバーが残りの作業量（画面数）認識ができなくなる場合があります。修正が入った場合に、全体への影響範囲（他の画面で同じ項目があれば修正が必要になる）が1画面づつしか見ていないためわからず、修正箇所の洗い出しに時間がかかってしまったり、最悪テストやリリースまで気づかないなどの、スケジュール遅延の原因にもなります。

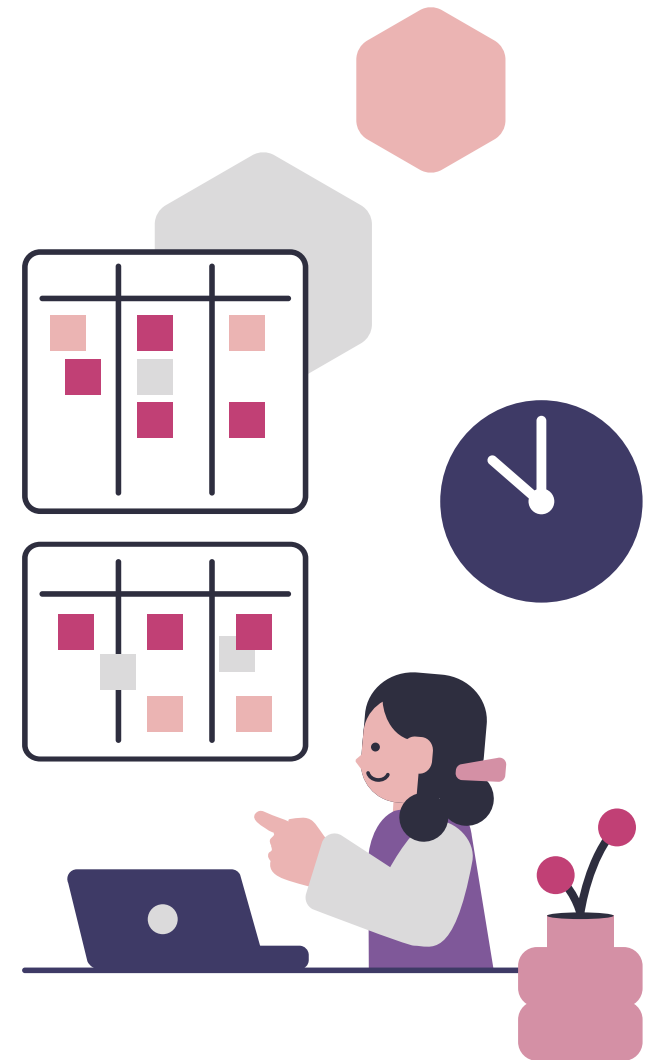
## 1-3. デザイナーの時間管理

締切はギリギリではなく早めに。  
できる人でも必ず修正は入ります。

締切までに何回修正できるかの逆算をしてから提出しましょう。（1→1.5ではなく、0→1を何回できるかです。）

デザイン締切は、必ずデッドライン日程（この日までにデザインがあがっていないと開発が進まない日付）を確認しすり合わせます。

自社開発の場合もそうですが、特に受託開発の場合、企業のブランドやマーケティング・法務など他部署のチェックが入ることがあるため、締切日程に担当者が確認しても間に合わない場合があります。





## 解答

問1 UIデザイナーのチームメンバーになる可能性がある人をすべて選択してください。

- ア. クライアント（依頼者）      イ. プロジェクトマネージャー  
ウ. プロダクトオーナー      オ. スクラムマスター  
カ. サーバーサイドエンジニア      キ. フロントエンドエンジニア  
ク. UXデザイナー      ケ. インフォメーションアーキテクト

**【解答】** ア、イ、ウ、エ、オ、カ、キ、ク、ケ、コ（全部）

- 
- クライアント（依頼者：自社開発の場合は内部、受託の場合は外部のお客様を含めた依頼者）もチームメンバーであることを意識して作成しましょう。

## 解答

問2 以下の文章を読み、正しいものをすべて選択してください。

- ア. デザイナーの仕事は、自分を表現することである。
- イ. デザイナーの仕事は、ユーザーがそのサービスを使用するときに、無理せず空気のように使用できるにはどうしたらいいかを考えることである。
- ウ. デザイナーの仕事は、見た目をの美しさのみを追求する仕事である。

### 【解答】 イ

- 自分を表現するのはアーティストであり、誰かの作りたいものや思いを形にするのが商業デザインです。自分の作品を制作しているわけではなく、誰かのためにデザインしています。
- カッコ良さや美しさを求めることは大切ですが、見た目「のみ」の追求では完成しません。ただし、機能のみが完成すればよいわけではありません。両者のバランスをとり、最適なデザインを提案することがプロの商業デザインです。
- ユーザーがゴールに辿り着けることを有しているのが商業デザインです。

## 解答

問3 木曜日のMTGでクライアントに4日後にデザインを提出してくださいと言われた場合、デザイナーとしての確認すべき項目をすべて書き出してください。（記述問題）

【解答例】クライアントの背景や、チームメンバーの状況などを想像して動くことが大切です。

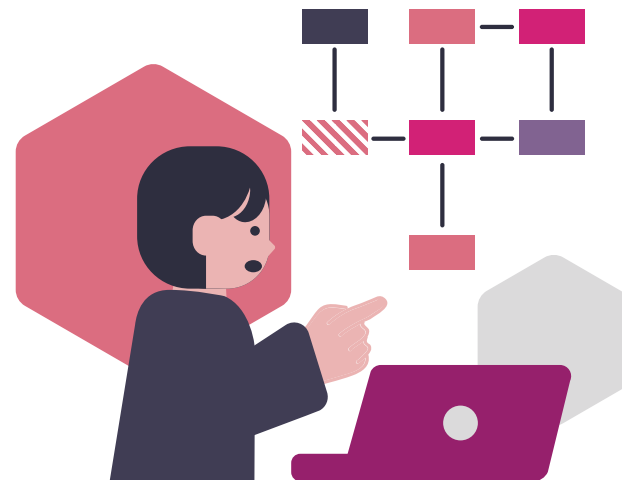
- クライアントのデザイン要望を確認する。話をしていると認識ずれなどもでてくるため必ず確認を行う。クライアントと表示内容は合意ができているか。その内容はエンジニアと合意が取れているか（フィジビリティチェックができているか）。
- クライアントは、デザインクオリティが完璧なものを見せてほしいのか。内容を確認できる程度の20%ぐらいの仮組みで見たいのか。またその理由はなにか？クライアントには、他にデザインを確認すべき相手が存在しているか？
- クライアントは土日を含む4営業日と考えた場合に、金月火水の4日とすると水曜日提出で問題ないのか？もしくは木金土日で、月曜日と言っているのか？。
- プロジェクトマネージャー・開発メンバーとはスケジュールを確認し、デッドライン日程を確認する。プロジェクトマネージャー・開発メンバーとは、4日後その後の校了で開発に間に合うか確認をする。もし、間に合わない場合には、スケジュール調整を行う必要がある。
- クライアントとは締切日を設定。確認MTGが必要であれば、日程調整を行う。

## PART 2 デザイナーの動き

クラスメソッドのUIデザイナーが

プロジェクト内でどのように動いているのか一例を記載しています。

これを見てUIデザイナーが実施にどんな仕事をしているのかイメージしましょう。



## この章の目的

1. プロジェクト内でのデザイナーの全体的な動きを理解する
2. サンプルをみて、自分ができる箇所、まだできない箇所を把握する
3. 自分だったらどう動くかを考える材料の一つとして流れを理解する

## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問4 以下の、空欄を埋めなさい。

要件ヒアリングで、最初に確認するのは  の  を実現したいのか  を明確にすることです。

## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問5 アジャイル開発について、正しい記述を選択しなさい。

- ア. クライアントの要望に応えるシステムをできる限り素早くリリースするという考えに基づく開発手法です。機能を最低限の状態にする方法が取られるケースが多く、開発メンバーがシステムにおける優先度に順位をつけ、短い期間での納品を目指して動きます。システムの優先順位を決めるために「スクラム」という手法が主に採用されます。
- イ. 開発を各工程に分けて開発を1つずつ確認しながら工程を進めていく手法です。クライアントに丁寧なヒアリングを行い、要件定義後に作成される基本設計を土台として詳細設計の工程へと進みます。そのため前の工程に欠陥があると次の工程へ進めません。また次の工程に進んでしまうと後戻りができない手法です。

## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問6 説明文の空欄を、以下の項目から選択しなさい。

ウォーターフォール開発において、デザイナーは最初に要件定義の締切日程を確認し、クライアントに他部署への  や  を確認してから、プロジェクトチームのメンバーまたはプロジェクトマネージャーと、デザインの  を確認を行い、デザインスケジュールの提示を行う。

- ア. スケジュール上のデザイン締切日    イ. 法務チェック日数    ウ. リリース日  
オ. 祝日など休日    カ. デザイン日数    キ. 確認の有無    ク. コンセプト  
ケ. デッドライン締切日    コ. ターゲット



## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問7 ワイヤフレームについて、正しい記述を選択しなさい。

- ア. ワイヤフレームとは、最終成果物とほぼ同じデザイン画面のことです。
- イ. ワイヤフレームは、項目や遷移、ボタンの配置などがざっくりと理解できるものです。
- ウ. ワイヤフレームは、「模型」の意味をもつ言葉で、裏側のシステムは実装されていないが、視覚的には完成形に近い動くサンプルを指します。

## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問8 以下の説明について、正しい記述を選択してください。

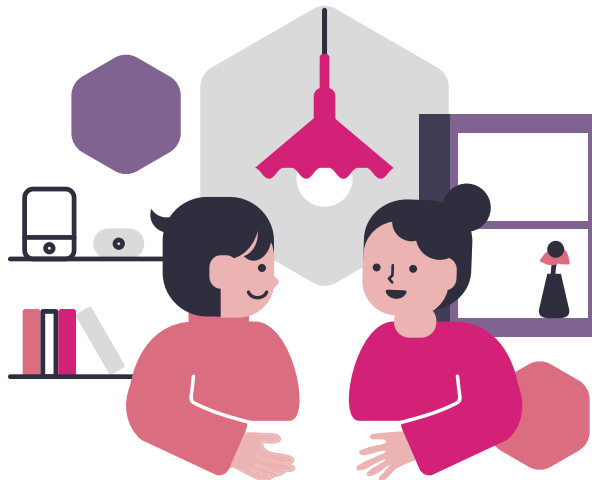
- ア. ワイヤーフレームからデザインのフェーズにおいて、デザイナーが納得するまでの100%デザインの状態ではなく、方向性がわかる程度の60%状態で、クライアントへ方向性を確認していくのが手戻りが少なくなる可能性が高い。
- イ. ワイヤーフレームからデザインのフェーズにおいて、デザインはデザイナーの責任範囲であるため、締め切りギリギリまで使って完璧なものを仕上げるのが良い。
- ウ. ワイヤーフレームからデザインのフェーズにおいて、デザインの範囲に収まらない要件が出てくる場合があるので、PMやエンジニアと連携をとって作業をしていくのが理想である。



段階での詳細を記載していきます。

## 2-1. プロジェクトの受注

受注が確定すると、アサイン調整が行われます。受注前のプリセールス時点から入る場合もありますが、正式な参加はここからです。



## 2-2. 要件ヒアリング

ビジネス要件の確認や、どのようなシーンで使用されるものかをクライアント（依頼者。受託の場合は社外。自社開発の場合は社内の依頼者）とチームメンバー（PM、エンジニア、UXデザイナー、UIデザイナー）で確認します。

- コンセプトの確認（何を実現したいか）
- 今開発における目標の確認（ゴールは？何が重要?）
- メンバーの担当範囲
- QCD（Quality（品質）、Cost（コスト）、Delivery（納期））の優先順位やバランス
- ウォーターフォール開発か、アジャイル開発かの確認。

## 2-3. 開発手法によるスケジュール確認

### ●ウォーターフォール開発

デザイナーは内部でのデッドライン（開発が始まる日）と、仕様が決定する時期の確認をし、クライアント側の社内確認などを有無を確認し、デザインのスケジュールを提示します。

プロジェクトマネージャーやリーダーがその役割を担う場合もありますが、自分の工数を算出するために必要な情報は把握しておきましょう。

### ●アジャイル開発（スクラム）

※スクラムの内容については割愛します。

クラスメソッドでは、デザイナーは「0スプリント」の実施の確認・調整を行います。

開発スプリントに入る前に、デザインのための要件決め0スプリントが入ることをチーム全員（クライアント（PO）・プロジェクトマネージャー（PM）・プロダクトマネージャー・スクラムマスター（SM）・エンジニア）と合意・確認しましょう。

（契約内容によっては実現不可の場合も存在します。）

## 2-3. 開発手法によるスケジュール確認

### ●アジャイル開発（スクラム）の補足

- 画面デザインまでを含めてスクラムでいうReady状態にする場合、「0スプリント」で開発の2週間前にはクライアントとのデザイン合意を行うことをお勧めします。企業側の知財やブランディング確認がある場合があり、十分な期間を確保していないと、Readyにすることができなくなります。
- ※「0スプリント」で合意後に、スクラムのプランニングフェーズで変更が入ることはあります。ただし、大枠のデザインは前もってOKを取っておきます。
- ※アジャイルはスプリントで画面が変わっていくため、エンジニアが最終デザインがわからない状態になる場合があります。今スプリントで使用するデザインは動かさずに、別データで次のスプリントの修正を走らせましょう。

## 2-4. 必要最低限の項目を洗い出す

ユーザーの利用シーンに合わせ、ユーザーが目的を達成するために最低限必要な項目をクライアントと確認します。ユーザーの行動シナリオやユーザーストーリーマッピングを使用しMVPを洗い出します。

(どのツールを使用するかはプロジェクトの性質によります)

※エンジニア参加者がいると、項目の表示データをもっているかを判断できたり、代替案がきける場合があるため、いっしょに確認を行う方が早い場合もあります。

※ステークホルダーに項目の優先順位をつけてもらえると、デザインはやりやすいでしょう。

例) ユーザーが商品を選ぶシーン

(現実世界を観察したり振り返ります。服を購入する時にどのような行動・思考をしてるのでしょうか？ POやUXデザイナーと協力し整理しましょう。機能優先になってしまいがちなので、デザイナーは目的からその機能が必要かという視点で考えましょう。)

	店に行く	店頭に並んだ商品を見る	気になる商品の詳細を見る	
行動 思考	ユーザーは 店を訪れたい	ユーザーはほしい 商品があるかみたい	ユーザーは商品を 吟味したい	
その目的を 達成するのに 必要な情報	ユーザーは ① (店に行くために) お店の場所を確認 できる	ユーザーは ① (ほしい商品があるか) 見た目を確認できる	ユーザーは ① (買うに値するか) 写真などで360度 商品を確認できる	ユーザーは ③ (自分のサイズ) サイズを確認できる
カッコ内の なんのためにを ヒアリングし 考えましょう。	ユーザーは ② (店に行くために) お店の外観を見ることが できる	ユーザーは ② (購入できる商品が あるか) 値段を確認できる	ユーザーは ② (購入できる商品か) 値段を確認できる	ユーザーは ④ (好きな色があるか) 他の色があるか 確認できる

地図 URL、電話など

検索・カテゴリーなど

## 2-5. ワイヤーフレームで全体像を確認

ワイヤーレベルで全体像を起こします。

※この段階では、全体の大きな流れを確認するため手書きの項目表示程度で問題ないです。ただし、ワイヤーを飛ばして画面を作成する場合があります。

- クライアントへ見せる前に社内の開発チームである、エンジニア・プロジェクトマネージャー (PM) ・プロジェクトリーダー (PL) と確認します。
- この項目を表示できるデータがサーバー側にあるかの確認も必要なため、サーバーサイドのエンジニアと話ができるとさらに良いでしょう。※細かいエラー処理などは、後で考えますので、ここでは正規のメイン動線を確認していきましょう。
- この段階で、情報設計・動線設計、アプリの骨格を決めていきます。

まず最初に、全体像をクライアントと確認しましょう。

店に行く

ユーザーは  
(店に行くために)  
お店の場所を確認  
できる ①

ユーザーは  
(店に行くために)  
お店の外観を見ること  
ができる ②

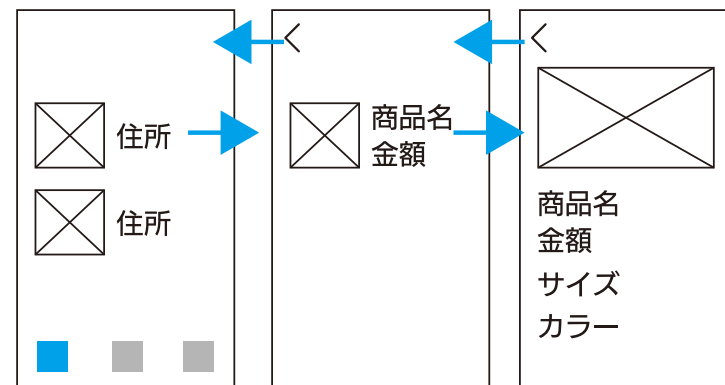
地図 URL、電話など

どんなユーザーが  
店に行きたいか？

必須は、  
・住所  
・外観

(このあたりを考  
えるとき、ユーザー像が必要)  
地図？っているかな  
現在地からの距離？

動線を繋げて行きましょう。画面全体を俯瞰できるように



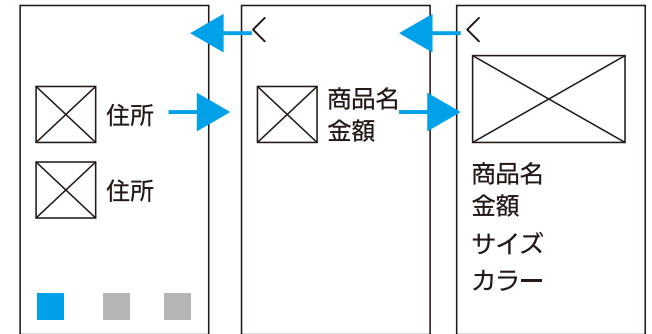


## 2-6. メイン機能の動線（目的達成まで）を再度情報設計・デザイン

目的ごとの要件・仕様を詰めていきます。

「〇〇ができる」ための一連の流れと、必要な機能をまとめる作業を開発メンバーで行います。

Figmaを映して、目の前でライブデザインしながらやる場合もあります。



店舗を検索し、店舗内の商品を確認できるまでの流れ。もう少し広げると、商品詳細確認まで



地図  
住所検索  
現在地からの距離

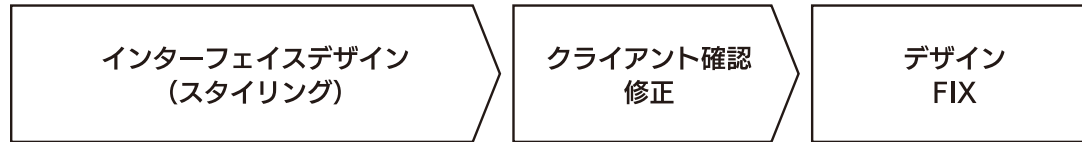


この情報で、ユーザーは商品を選べるか？  
カテゴリーフィルターは必要か？  
均一に並べるのか？  
説明は必要か？写真が大きい方がいいのか？  
実装は？



## 2-7. 1機能ごとディテールを詰めて確認していく

仕様が決まったら、一連の機能を1機能ごとに画面の流れでデザインしていく作業に入ります。



### 2-7-1. インターフェイスデザイン (スタイリング)

(1) 初期に各種スタイルを決定していきます。設定したものだけしか使用してはいけないという話ではないのですが、最低限の設定で作るために初期に枠組みを作ります。

#### a. 色スタイル設定

- 多色の使用は、統一性がなくなること、デザイナーが把握できなくなるデメリットがあります。
- 視認差の微妙な色を無限に作らない予防としてある程度使用する色を最初に決めます。
- 目安としてグレースケール5~6色、メインカラー・サブカラー・アクセント程度です。

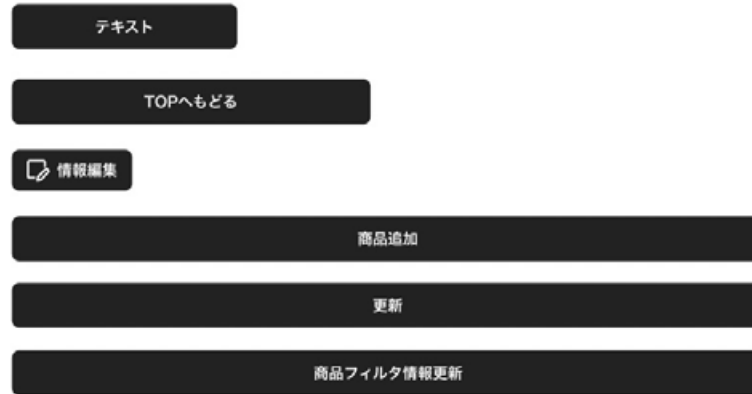
#### b. 文字スタイル設定 (タイポグラフィ)

1機能ごととは  
「会員登録ができる」  
機能例

- 「新規登録画面→入力フォーム画面→確認画面→プライバシーポリシーなど→登録完了ボタン」が1機能。
- 上記のような一連の画面が完成したら、エンジニアに説明します。
- 他デザイナーへデザインレビューを行ってから、エンジニアやPMへ確認する場合もあれば、エンジニアやPMとライブデザインをしながら、大体の遷移を決めていくこともあります。ただし、デザイナーの熟練度が必要になります。

## Buttons

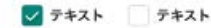
btn\_primary



Form



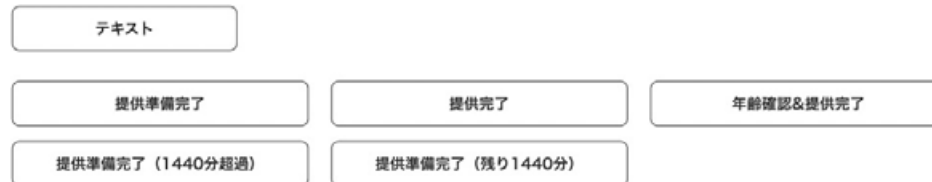
CheckBox



RadioButtons



btn\_secondary



- (2) 1画面ずつ取り掛かります。同じようなデザインが出てきた場合、全体の統一コンポーネントを考えながらデザインします。
- (3) 社内でレビューを行ったら、その部分を修正してもう一度社内確認を行います。
- (4) 社内でOKがでたら、クライアントへ確認を行います。

## 2-7. 1機能ごとディテールを詰めて確認していく

### 2-7-2. クライアントの確認

- (1) ワイヤフレームでも確認しましたが、スタイルを当てた最初の確認は個別デザインではなく、メイン機能の一連の動きに対して、OKをもらうようにしましょう。元々テキストで確認していた内容がビジュアルになることで解像度が上がり、そこではじめて気付くことが多くあるためです。ビジネス都合でどうしても作業しなければならない場合も多くあるため、クライアントとの確認と調整はここで徹底します。

※表面デザインの修正はデザイナーのスタイリングの話になりますが、表示項目の変更は、裏側のシステムの問題もあるため、持ち帰ってエンジニアやPMと再度検討する必要があります。早めの確認が大切です。

※ワイヤフレームで項目は確認していても、実際の画面デザインができあがると想像できていなかった部分がクリアになり、修正要望が発生します。

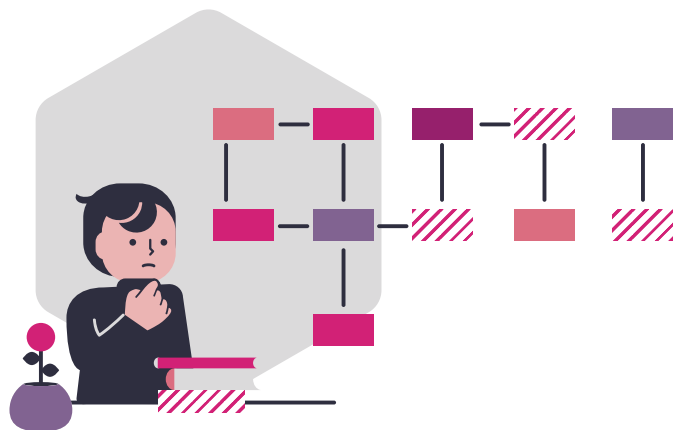
- (2) 確認段階で、ビジネス視点や開発側で見落とししてしまったユーザーの価値感や会社のブランディングなどのフィードバックが入ってきます。ここでもユーザーストーリーと目的を明確にし、フィードバックがどのような背景からきているのか「そもそも」を考えてデザインします。

ウォーターフォールの場合は、この時点での項目の変更は仕様変更となりスケジュールの再調整となりますので調整が発生します。アジャイルの場合は、この時点での変更も対応可能ですが、どちらにしてもスケジュールち内容の再確認・検討は必要になります。

(3) 動線OKになったら、次にデザインの方向性がOKか否かを確認しましょう。

大枠のデザインOK後に1画面ずつをブラッシュアップといって細かな部分をデザインしていきましょう。最初から小さな部品を修正していくと、進まなくなります。

●方向性があるていば、1機能分の画面を2~3回調整すると、必要コンポーネントがそろいます。他機能の画面は作成したコンポーネント流用でデザインしていくことが可能です。



●方向性が違う場合、メイン機能の1~2画面を2~3案作成したりしてスタイルを確認していきます。

2回ほど確認してダメな場合、根本的に間違っている場合があるため、ブランディング担当者様と直接MTGをセッティングしましょう。

その際、何を参考にされているのか？（ベンチマーク）をMTGまでに集めていただいたり、現在の他のサービスECやクライアント企業様のデザインを収集したりします。

（自信がない場合は、先輩デザイナーやメンターなどに相談しましょう。）

(4) 一個ずつ着実に進めていきます。

※どんなにすごいデザイナーでも1発OKはありません。

(5) デザインFIXを合意します。

FIX後には『必ず』開発が開始したら今後デザインの大幅な変更はできないことをクライアントと再度確認します。

「作成する前に仮説の価値検証」を行えるラストチャンスです。動くプロトタイプは時間がない場合もあるため、簡易的に壁にデザインを順番に貼って「確認する」などでもよいです。今回のゴールは達成できますか？をメンバーと見ましょう。

グラフィックソフトで修正することは簡単ですが、アプリ開発の場合は、実装またはバックエンドの開発が必要になるため仕事で行う以上は、Quality（品質）、Cost（コスト）、Delivery（納期）がどうしてもトレードオフになります。

※デザイナーとしては、良いものを作りたいクライアントの要望には共感できます。

しかし「それがないと価値が出せないのか」はチームで考えるべき事項です。

印刷物も一度印刷してしまうと、修正は再印刷がかかりコスト高になるのですが、アプリも同じで、エンジニアの工数とストア申請が必要なため慎重にいきましょう。

ストア申請前であれば対応できるケースもあります。

例えば、バックエンドが関与しない文言の修正程度であれば対応可能です。

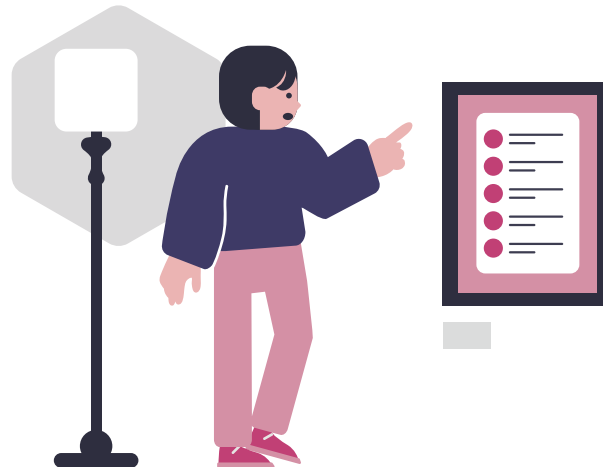
それ以外のバックエンドから返しているテキスト修正や新しい項目の追加はスケジュールとの調整となりますので、チームに相談・一緒に考えることを頭に置いておきましょう。

（チーム開発は、デザインの世界だけで終わりません）

(6) デザインFIX後、開発がはじまると開発開始前には判明しなかった問題が発生しますので、そちらのデザイン対応を行います。

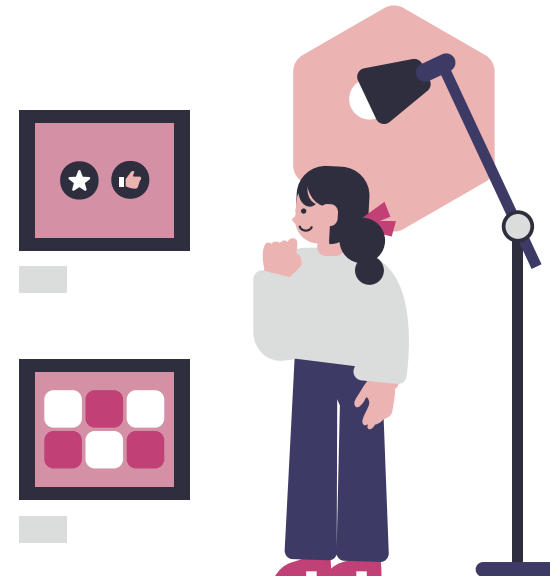
対応例)

- a. システム上のAPIエラーが発生することになるので、画面の構成を変更する必要がある。
- b. 法務チェックなどからテキストの文言修正やもしくは、画面デザインそのものを変更することもあります。



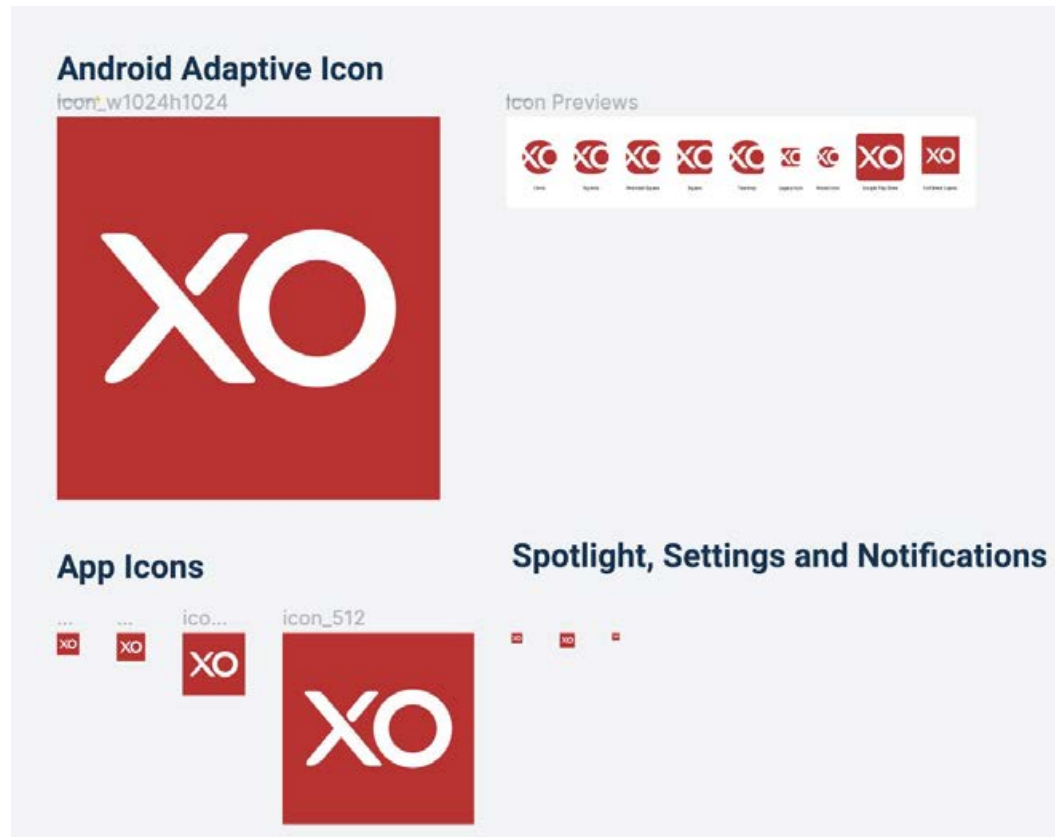
(7) テストバージョンで実装されたUIのデザインチェックを行います。

- a. テキストのミス
- b. 色の指定の確認
- c. 余白のサイズの確認
- d. 触った感じのインタラクション



(8) テストが公開されたあたりから、ストア申請作業のデザインタスクが発生する場合があります。

- a. アプリストア用のホーム画面アイコン作成
- b. ストア用のスクリーンショット作成





(8) 無事にアプリがリリースしたら、初期開発は完了です。

その後、保守対応があったり納品項目にデザインデータや、画面遷移図、画面設計書などがある場合は納品が発生したりします。

次期開発において担当デザイナーが変わる場合もありますので、デザインデータは最終デザインFIXのデータでわかりやすくしておきましょう。

- 使用していないレイヤー・イメージ・アイコンの削除
- レイヤー名・アートボード名の名前の整理（copyはやめましょう）
- 使用しているカラー設定を適用させるなど

半年後の自分がわかるように整理する気持ちでやるのがコツです。大抵は覚えていないものです。（未来の自分への投資）



## 解答

問4 以下の、空欄を埋めなさい。

要件ヒアリングで、最初に確認するのは  の  を実現したいのか  を明確にすることです。

**【解答】** ① ユーザー    ② 何    ③ ゴール

---

## 解答

問5 アジャイル開発について、正しい記述を選択しなさい。

- ア. クライアントの要望に応えるシステムをできる限り素早くリリースするという考えに基づく開発手法です。機能を最低限の状態にする方法が取られるケースが多く、開発メンバーがシステムにおける優先度に順位をつけ、短い期間での納品を目指して動きます。システムの優先順位を決めるために「スクラム」という手法が主に採用されます。
- イ. 開発を各工程に分けて開発を1つずつ確認しながら工程を進めていく手法です。クライアントに丁寧なヒアリングを行い、要件定義後に作成される基本設計を土台として詳細設計の工程へと進みます。そのため前の工程に欠陥があると次の工程へ進めません。また次の工程に進んでしまうと後戻りができない手法です。

### 【解答】 ア

- 各工程で確認していくのはウォーターフォール開発です。どちらもメリット、デメリットが存在し、どちらがよいということはありません。

## 解答

問6 説明文の空欄を、以下の項目から選択しなさい。

ウォーターフォール開発において、デザイナーは最初に要件定義の締切日程を確認し、クライアントに他部署への  や  を確認してから、プロジェクトチームのメンバーまたはプロジェクトマネージャーと、デザインの  を確認を行い、デザインスケジュールの提示を行う。

- ア. スケジュール上のデザイン締切日    イ. 法務チェック日数    ウ. リリース日  
 オ. 祝日など休日    カ. デザイン日数    キ. 確認の有無    ク. コンセプト  
 ケ. デッドライン締切日    コ. ターゲット

**【解答】** ① キ    ② イ    ③ ケ

- ウォーターフォール開発の場合は、日程や確認事項が大切になります。特にデザイナーの場合は、社外のリリース日などの締め切りと、開発内部でのデザイン締め切りがあるため、混乱しないように認識しておきましょう。※①、②の解答については、他にも考えうる解答があるかもしれません。自分で考えて、何が必要なのか想像しましょう。

## 解答

問7 ワイヤフレームについて、正しい記述を選択しなさい。

- ア. ワイヤフレームとは、最終成果物とほぼ同じデザイン画面のことです。
- イ. ワイヤフレームは、項目や遷移、ボタンの配置などがざっくりと理解できるものです。
- ウ. ワイヤフレームは、「模型」の意味をもつ言葉で、裏側のシステムは実装されていないが、視覚的には完成形に近い動くサンプルを指します。

### 【解答】 イ

---

- ビジュアルデザインが、最終成果物とほぼ同じデザインです。
- ウは、プロトタイプやモックアップのことです。

## 解答

問8 以下の説明について、正しい記述を選択してください。

- ア. ワイヤフレームからデザインのフェーズにおいて、デザイナーが納得するまでの100%デザインの状態ではなく、方向性がわかる程度の60%状態で、クライアントへ方向性を確認していくのが手戻りが少なくなる可能性が高い。
- イ. ワイヤフレームからデザインのフェーズにおいて、デザインはデザイナーの責任範囲であるため、締め切りギリギリまで使って完璧なものを仕上げるのが良い。
- ウ. ワイヤフレームからデザインのフェーズにおいて、デザインの範囲に収まらない要件が出てくる場合があるので、PMやエンジニアと連携をとって作業をしていくのが理想である。

### 【解答】 ア、ウ

- ギリギリまで手元にもっていた場合、クライアントの要望にそったものを出せるかどうかのギャンブルになってしまいます。  
また、一度の確認でOKがあることはありません。なるべく早い段階からチームメンバーやクライアントと確認し進めることを推奨します。



## PART 3 デザイン前の前提知識

アプリをデザインするその前に、大切な知識を記載しています。

## この章の目的

1. デザインをする上での著作権の知識を知る



## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問9 以下の説明について、正しい記述を選択してください。

- ア. 自社で契約した商用利用可サイトからダウンロードした素材を自社のPR広告に使用した。
- イ. 自社で契約した商用利用可サイトからダウンロードした素材をA社の案件で使用した。
- ウ. WEBで検索した写真素材を、自社の広告に使用した。
- エ. WEBで検索した写真素材を、B社の広告に使用した。
- オ. 自社で契約した商用利用可サイトより、人物素材を使用した。

## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問10 以下の説明について、どんな問題があったか。またどのような対応が必要か記述してください。

### 【説明】

クライアントからA社のアプリをベンチマークしているのですが、全く同じに作成してほしいと依頼があり、ベンチマークアプリをスクリーンショット、トレースして作成を行った。

## PART 3：デザイン前の前提知識

### 3-1. 著作権、意匠権、ライセンス

商業デザイナーにとって大切な著作権の話から記載します。

専門家の意見が必要になりますので、企業様の法務チェックや弁護士さんなどにお任せすることになる問題です。

ただし、UIデザイナーとして知っておくべき知識が存在しますので、心に止めておきましょう。

デザイナーが注意を払わなければならない項目

- 3-1-1. 無料素材の使用
- 3-1-2. WEBで検索したイメージ画像・写真の使用
- 3-1-3. 写真イメージ素材・音素材のライセンスの所在
- 3-1-4. これと同じ画面を作成して欲しいと言われたら？

### 3-1-1. 無料素材の使用

著作権フリーのアイコン・イメージ、また無料でダウンロードできる素材が存在します。

プロの結論としては、場合によりますが、なるべく『使用しないがよい』です。特に無償素材については、仕事として依頼された場合は使うことは控えましょう。

#### 【使用しないほうがよい理由】

(1) 他の会社のデザインと同じになる可能性がある

アプリは数多く存在します。その中で差別化を図る場合に、同じアイコンを使用しているとは同じ印象になってしまう可能性が高くなります。

(2) 著作権問題が発生する可能性がある

フリー素材は、ライセンス表示が必須の場合があります。トップに必ず記載が必要などの対応に「企業アプリでその対応が可能か」の問題が発生します。

もしもライセンス表示をよく読まずに使用したり、無料素材が有料化されたりした場合に訴えられるのは誰になるのかを考えると、所属している会社（クラスメソッド）、またはクライアントとなります。有名企業であれば更に大きな問題に発展します。

ライセンス記載を行えば問題ないということではなく、その対応を考えないためにデザイン工数を支払いいただいている場合もあるため、アプリ内で使用するアイコンはWEBからダウンロードしてくるのではなく、作成する方向で作業しましょう。

### 3-1-2. WEBで検索したイメージ画像・写真の使用

あえて「強い禁止事項」とします。

企業としてデザインの仕事をする上で、絶対に使用してはいけません。

こちらの場合は、完全に著作権などの法的な問題に関わります。

企業でのデザイン仕事は、世の中からはクラスメソッドが出しているという認識で見られるものです。

そのためデザイナー個人が行った行為でも、クラスメソッドが責任を問われることとなります。

では、参考にしたいデザインがある場合にどうするかですが、参考にするとどめます。

同じようなイメージを使いたい場合は、以下で対応します。

- 有料サイトで購入する
- プロのカメラマンに依頼する
- 自分で作成する

### 3-1-3. 購入した写真イメージ素材・音素材のライセンスの所在

有料の写真素材や音素材に関する記載をします。Adobe Stockなどの有料サイトで購入した素材のライセンスの所在の話です。

『ライセンスは、公開する企業の名義で購入しましょう。』

A社が購入した素材のライセンスは、A社が公開するアプリで使用するのであれば問題ないのですが、B社のアプリで使用することはできません。B社のアプリで使用する場合は、B社でライセンス料金を支払います。

また、注意事項は読み飛ばさずに必ず全部を確認しましょう。

特に人物の入った写真などは要注意です。モデルさんの肖像権の期間なども入ってきますので、利用規約は最後まで読みましょう。

●「ロイヤリティフリー」  
著作権に保護されている作品は、使用する側が著作者に使用料（ロイヤリティ）を「毎回」支払う必要があり、その「毎回支払う使用料」を無料にしたものが「ロイヤリティフリー」です。利用可能な範囲内での使用します。

●「著作権フリー」  
サイトにより若干違いがあるのですが、素材に著作権がないわけではなく、作品の制作者（著作権者）が著作権を行使しないことを意味しているものです。ロイヤリティフリーと同じような意味合いでも使われます。利用可能な範囲内での使用します。

●「ライツマネージド」  
使用する期間などの取り決めのある素材。

●「商用利用可」  
ビジネスで利用できる素材。

### 3-1-4. これと同じ画面を作成してほしいと言われたら？

全く同じものを作成することは、プロのデザイナーとして肯定はできません。

訴訟などが起きた場合に、同じ画面をコピーしたとなると問題になります。

自社開発の場合は自分の会社、受託開発の場合はさらに依頼した会社がその対応を行うことになります。

デザイナー個人の問題ではなくなってしまふことを意識しましょう。

ユーザーゴールにむけて考えた結果、同じような画面になることはありえますが、その過程がなくコピーすることはプロとしてやってはいけません。

## 3-2. 知っておくとよい知識

### 3-2-1. 現状のアプリの画面デザインと著作権

『独自性が特に強い部分を完全コピーした場合は著作権侵害となる』

※ここに記載してあることは、知っておくとよい知識であり、本格的には法律の専門家への相談が必要となるものです。

著作権侵害が認められるためには、A社のアプリからB社のアプリの表現上の本質的特徴を直接確認できることが必要となります。

抽象的なアイデアが類似していたり、機能に由来するデザイン部分が共通であるとしても、著作権侵害にはならないようです。

独自性が高い画面デザインでなければ、画面の類似性はある程度存在しているということであり、独自のアイデアや創作性のない表現が共通する場合は著作権は成立しない状態です。

ただし、現状なため将来の裁判で違う判決となる可能性もあります。



## 3-2. 知っておくとよい知識

3-2-2. アプリ・ソフトの『画面表示：UI』は『実用的』のため『創造性』が否定されやすい

※ここに記載してあることは、知っておくとよい知識であり、本格的には法律の専門家への相談が必要になります。

UIはそれぞれのアプリ・ソフトの種類・機能によって異なりますが、

『実用性』という共通の目的があることから、一定の傾向があることは一般的に認識されています。

裁判の傾向は『機能が同じアプリなら似ていて当然＝違法ではない』という人が多いようです。

特定の表示・表現・動的仕組み発想	創造性	著作物の判定
1人しか思い付かない	あり	肯定する方向性
数人以上が思い付く	なし	否定する方向性

ただ、現状がそうであるだけということは認識しておきましょう。

### 【システム画面デザインの判例】

スケジュール管理ソフトウェアの画面デザインについて著作権侵害の成否が争われ、否決されたとして、東京地裁平成14年9月5日判決があります。

著作権侵害（複製権・翻案権侵害）の対象

- デッドコピー（完全なコピー）ないしそれに準ずるようなものに限られる
- 画面全体における『共通する表現』と『異なる表現』の割合も影響する

### 3-2-3. UI・画面表示『意匠』

※ここに記載してあることは、知っておくとよい知識であり、本格的には法律の専門家への相談が必要になります。

UI・画面表示は『意匠』に該当することもあります。

意匠法で『画面表示』についても対象として規定されています。

意匠法2条1項 2項

[https://www.jpo.go.jp/system/laws/rule/guideline/design/shinsa\\_kijun/kaitei/document/h18\\_isyou\\_kijun/setsumeikai01.pdf](https://www.jpo.go.jp/system/laws/rule/guideline/design/shinsa_kijun/kaitei/document/h18_isyou_kijun/setsumeikai01.pdf)

アプリ・ソフトにおけるUIは、『意匠』としては認められない傾向が強い。

仮にこの点をクリアしたとしても、登録要件の1つ『新規性』などが欠けているとして、意匠権として認められない可能性も高いとなっています。

この辺りは、もっと詳細となると専門家への相談しましょう。

## 解答

問9 以下の説明について、正しい記述を選択してください。

- ア. 自社で契約した商用利用可サイトからダウンロードした素材を自社のPR広告に使用した。
- イ. 自社で契約した商用利用可サイトからダウンロードした素材をA社の案件で使用した。
- ウ. WEBで検索した写真素材を、自社の広告に使用した。
- エ. WEBで検索した写真素材を、B社の広告に使用した。
- オ. 自社で契約した商用利用可サイトより、人物素材を使用した。

### 【解答】ア

- 
- A社で使用する場合は、A社でライセンスを取得します。
  - WEB検索だけでは、基本的に商用利用可能なか個人の写真なのか判断が付きません。必ず確認しましょう。
  - 人物素材には、モデルリリースという契約があり、その内容を確認しないで使用することはやめましょう。

## 解答

問10 以下の説明について、どんな問題があったか。またどのような対応が必要か記述してください。

### 【説明】

クライアントからA社のアプリをベンチマークしているのですが、全く同じに作成してほしいと依頼があり、ベンチマークアプリをスクリーンショット、トレースして作成を行った。

- トレースしての完全コピーはデッドコピーといわれ問題があります。
- ユーザーの行動や目的から考えていないため、プロジェクトにおいての価値を提供できているかはわかりません。
- デザイナーは法律の専門家ではないですが、クライアントへ模倣やライセンスに関する知識をお伝えしたり、信頼おける法律の専門家にご相談されるように促すことも時には必要となるかもしれません。（PMがやるべきとか、クライアントがしておくべきとかそういったことではなく、自分毎として知っておきましょう）

※こちらにもいろいろな考え方があります。知識として知っている必要はありますが、専門家にお任せしましょう。

## PART 4 アプリデザイン基礎知識

デザイナーの役割のうちのひとつ、  
アプリケーションUIの部分の基礎知識を説明していきます。



## この章の目的

1. 公式ガイドラインを知る
2. 実際に作る上での基礎知識を知る
3. OSに対応したデザインを理解する

## まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問11 以下の、空欄を埋めなさい。

スマートフォンアプリの開発において、代表的なデザインガイドラインがあります。

iOSは  、Androidは  が存在します。

まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問12 スマートフォンの解像度について、正しい記述を選択しなさい。

- ア. スマートフォンの解像度は1インチの正方形の中にどれだけドットが存在するか「密度」で表します。
- イ. スマートフォンの解像度は、画面上にあるピクセルの数量で表します。



まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問13 iOSの素材を書き出す場合の形式を、すべて選択してください。

ア. ai    イ. psd    ウ. pdf    エ. png    オ. svg    カ. jpg    キ. gif

まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問14 Andoroid 素材を書き出す場合に必要な倍数を、以下より選択してください。

- ア. 0.75 倍    イ. 1.0 倍    ウ. 1.5 倍    エ. 2.0 倍    オ. 3.0 倍  
カ. 4.0 倍    キ. 5.0 倍    ク. 6.0 倍

まずは現状の理解を確認します。

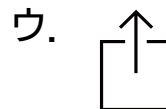
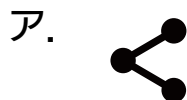
説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問15 Android のバックボタンの挙動を記述してください。

まずは現状の理解を確認します。

説明を読む前に以下の問題を解いてみましょう。その後説明を読み再度解答してください。

問16 iOSのシェアアイコンを選択してください。



## PART 4：アプリデザイン基礎知識

### 4-1. 最初はOSの違いを理解しましょう

スマートフォンは代表的な2つの OS があります。Apple 製品の OS である iOS と、その他スマートフォンに搭載されている Android です。

それぞれ、Apple と Google が提供している公式のデザインガイドが存在し、およそ1年で公式のアップデートが行われます。

小さな更新をされていたりしますので、アプリを制作するときに、確認しながら作成することを推奨します。

OS	デザインガイドライン	URL
iOS	Human Interface Guidelines	<a href="https://developer.apple.com/design/human-interface-guidelines/">https://developer.apple.com/design/human-interface-guidelines/</a>
Android	Material Design	<a href="https://material.io/">https://material.io/</a>

また、公式がだしているテンプレートがありますので、resourceからダウンロードして活用することも可能です。

必須ダウンロードはフォントです。

OS	フォント	内容
iOS	San Francisco フォント	iOSの標準フォントです。日本語は標準ヒラギノフォントを使用します。20pt以上ではSF Pro Display、それ以下ではSF Pro Textを使用します。
Android	Roboto、NotoSans	Androidは機種によって様々なフォントが存在するのですが、基本はRoboto、Noto Sans JP、モトヤマルベリを使用します。

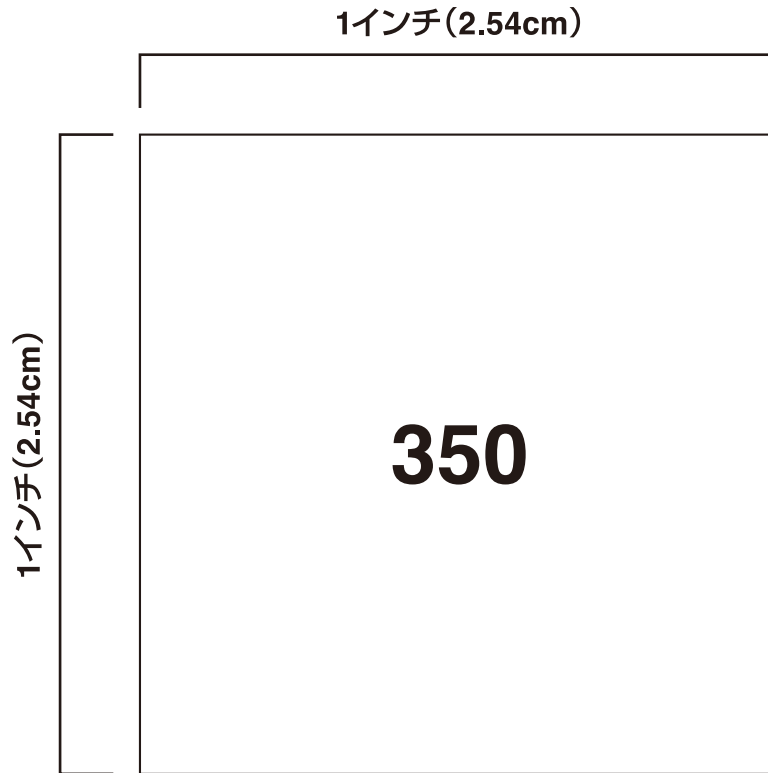
その他、参考になるデザインガイド

名称	内容
Windowsのデザインガイド	<a href="https://docs.microsoft.com/ja-jp/windows/apps/design/">https://docs.microsoft.com/ja-jp/windows/apps/design/</a>
IBM Design Language	<a href="https://www.ibm.com/design/language/">https://www.ibm.com/design/language/</a>
FUJITSU Transformation by Design (日本語版)	<a href="https://www.fujitsu.com/downloads/JP/about/businesspolicy/tech/design/article/document/transformationbydesign-jp.pdf">https://www.fujitsu.com/downloads/JP/about/businesspolicy/tech/design/article/document/transformationbydesign-jp.pdf</a>

## 4-2. アプリと解像度の理解

### 4-2-1. 印刷 (DTP) の解像度

350dpiは1インチの正方形の中にどれだけドットが存在するのかの「密度」で表します。



1インチ(2.54cm)あたり  
350個のドットでできているので解像度350  
印刷の場合はこちら

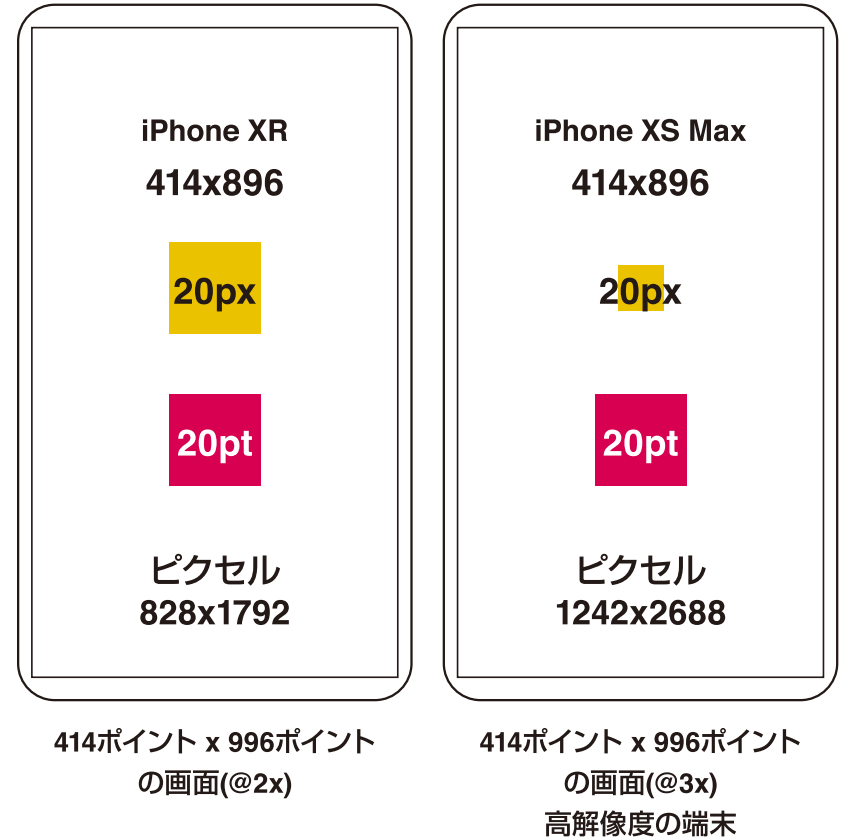
## 4-2-2. WEBやスマートフォンの解像度

解像度は密度ではなく『画面上にあるピクセルの数量』を表します。

重要なのは【端末ごとにピクセルのサイズが違う】ものが存在することです。ピクセルサイズが違うため、単位をピクセルで指定すると、高解像度の端末の場合にサイズが小さくなってしまいます。

Adobe Illustrator、Sketch、Figmaなどのベクターデータを扱えるソフトウェアでは、1倍サイズで制作を行い素材の書き出し時に、@2xや@3xを書き出すことが主流です。

Adobe Photoshopなどのラスターデータを扱うソフトでは、最大サイズ（2022年現在、iOS:3倍、Android:4倍）で作成します。





## 4-3. プラットホームに応じたデザイン知識

### 4-3-1. iOS と Android の単位

iOS の単位は「pt」です。

現在 @2x と @3x の端末のみしか存在しないため、1倍は作成せずに2倍と3倍の大きさのみを書き出します。

解像度	倍率	pt	px
@1x	1.0 倍、基準	1pt	1px
@2x	2.0 倍	1pt	2px
@3x	3.0 倍	1pt	3px

Android の単位は「dp」です。テキストには「sp」を使用します。

現在低密度サイズ (ldpi) の端末は存在しておらず mdpi から xxxhdpi サイズです。

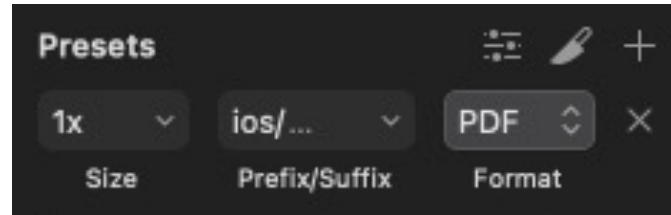
解像度	倍率	dp	px
低密度 (ldpi)	0.75 倍	4dp	3px
中密度 (mdpi)	1.0 倍、基準	4dp	4px
高密度 (hdpi)	1.5 倍	4dp	6px
超高密度 (xhdpi)	2.0 倍	4dp	8px
超超高密度 (xxhdpi)	3.0 倍	4dp	12px
超超超高密度 (xxxhdpi)	4.0 倍	4dp	16px

画面互換性の概要 (Android公式)

<https://developer.android.com/training/multiscreen/screendensities?hl=ja>

## 4-3-2. iOSの素材書き出し

### a. 単色の場合はSVGかPDF

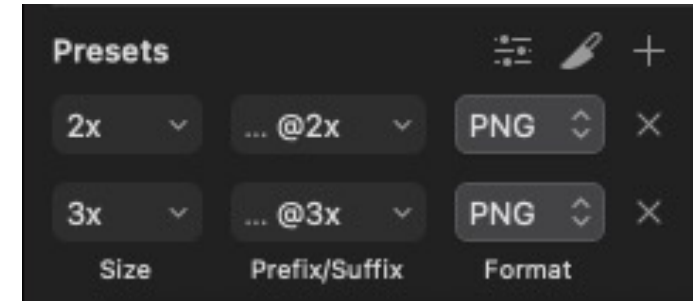


単色のアプリ内アイコンなどの場合、クラスメソッドではSVGもしくはPDFで書き出しを行います。書き出し形式は、フロントエンドエンジニアと確認すると間違いありません。

※ iOS は Xcode12 から SVG 対応しています。最新のものであれば、Android と合わせて SVG で書き出すのもあります。

※ 複雑な形のアイコンの場合、原因不明で表示できない場合もありますので、フロントエンドエンジニアと連携し、画像書き出しで対応する場合があります。

### b. 複数の色を使用している場合はpng形式

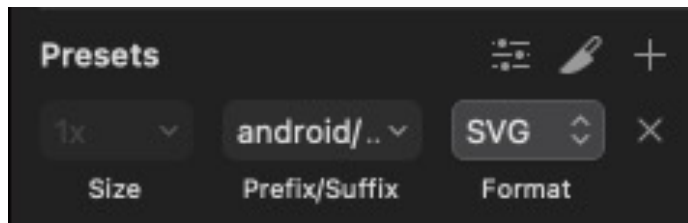


複数の色を使用している場合は、pngで書き出します。「Assets/ios/同じ素材名 @2x.png」のようなファイル名として作成することが多いです。

※ なるべく軽い素材にします。

### 4-3-3. Androidの素材書き出し

#### a. 単色の場合はSVGファイル



ソフトウェアによりますが、書き出しの場合は余計なデータが入るものも存在するため

<https://jakearchibald.github.io/svgomg/>

などでデータ自体を軽くしておくことが後々のアプリ容量にも関わってきます。

※ 複雑な形のアイコンの場合、原因不明で表示できない場合もありますので、フロントエンドエンジニアと連携し、画像書き出しで対応する場合があります。

#### b. 複数の色を使用している場合はpng形式



複数の色を使用している場合は、png画像の1倍、1.5倍、2倍、3倍、4倍で書き出します。Android「Assets/android/フォルダ名/同じ素材名.png」の形でフォルダ名に「Assets/mdpi/同じファイル名.png」をAssetsファイルとして作成することが多いです。

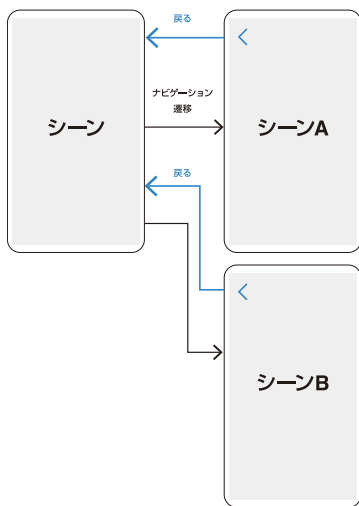
#### 4-3-4. 知識としてのページ遷移の概念（過去と現状）の違い

ナビゲーションの原則（Android公式）

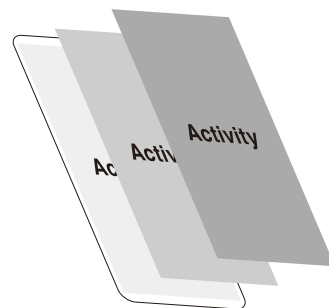
<https://developer.android.com/guide/navigation/navigation-principles?hl=ja>

iOSの場合はシーンを行き来し、  
Androidは新たな画面を上重ねていく  
イメージです。

iOS



Android



最近のAndroidは、Activity重ねることが減っていたり、実装における自由度が高まっているため、一概に断定はできません。（現状はひとつのFragmentを書き換えるが主流）

主流が切り替わる変化が早いこともあり、これが絶対ということはありません。

各ページ間の親子・兄弟関係を意識して遷移を考え、フロントエンドエンジニアとコミュニケーションをとることがデザイナーにとって大切な軸になります。

#### 4-3-5. Androidの標準ボタンと戻るボタン

- 適切な「戻る」ナビゲーションを提供する (Android公式)

<https://developer.android.com/guide/navigation/navigation-custom-back?hl=ja>

Androidは標準ボタンというものが存在し、「戻る (バック) ボタン」が存在します。見た目はAndroid端末ごとに違います。ジェスチャーナビゲーションで見えないボタンが実装されていることもあります。

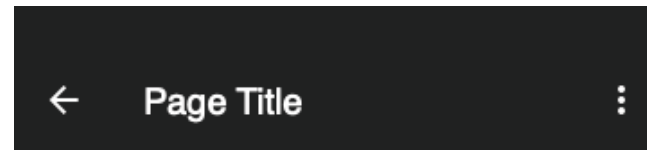
バックボタンはアプリを超えて「ひとつ前のアクティビティに移動する」ものです。上へボタンは「アプリ階層を上げるため」に使用します。

デザイナーはバックボタンの挙動と、「戻る」ボタンや「上へ」ボタンを配置したときの戻る画面の挙動を考える必要があります。(この時、Androidエンジニアと話をするとよいでしょう。) iOSの場合、標準ボタンがないので、ナビゲーションバーに遷移前の画面へ戻るボタンが存在しています。

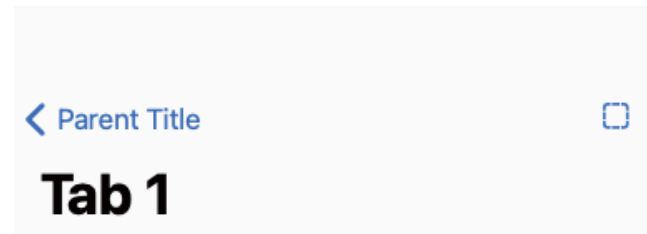
Android : 戻る (バック) ボタン



Android: 「←」上へ (アップ) ボタン



iOS: 戻るボタン



#### 4-3-6. 画面を止めるモーダル表示

- Human Interface Guidelines > Modality

<https://developer.apple.com/design/human-interface-guidelines/ios/app-architecture/modality/>

- Material Design > Dialogs

<https://material.io/components/dialogs>

画面をとめてユーザーに明示的行動を促すモーダル表示は、iOSの概念で実はAndroidには存在していません。

Androidではダイアログという概念で実装されます。

画面を止めてユーザーに明示的行動を促すは、決裁確認など重要な部分でよく使われます。

「ユーザーが重要な情報を受け取り、必要に応じてそれに基づいて行動できる」ように、確認、アラートなどユーザーに判断を委ねる場所で使用します。

Androidダイアログ



#### 4-3-7. ラジオボタン

iOSには、実はラジオボタンの標準UIは存在しません。

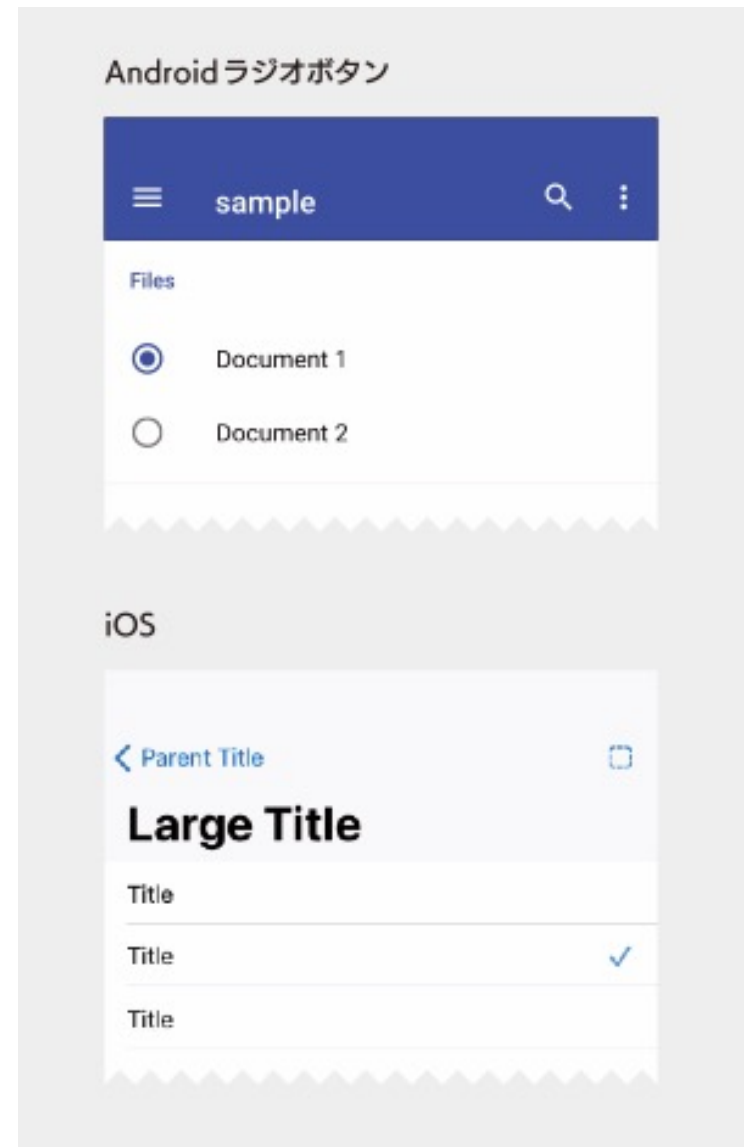
該当するものとして、チェックマークのアイコンが用意されています。選択肢をタップすると、右側にアイコンが表示されるものです。

ラジオ画像を用意することで、表現しているアプリも多く存在します。

その場合はアクティブと非アクティブの画像を用意します。

デザインの意味で考えるのであれば、排他制御（項目内のどれか一つのみが選択されること。Document1を選択するとDocument2は選択されない状態）にはラジオボタンを使用し、複数選択可能な場合にチェックボックスを使用します。

ユーザーに親切なのは、画像を用意してでもラジオボタンとチェックボックスを分けるデザインです。



#### 4-3-8. チェックボックス

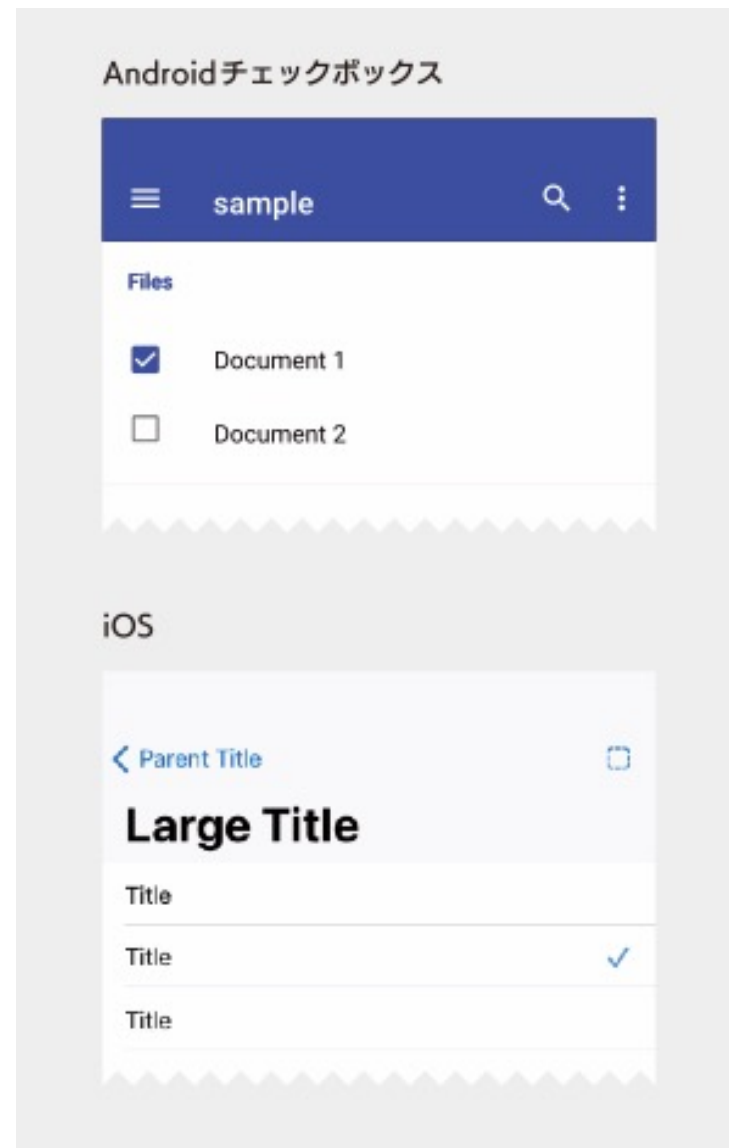
ラジオボタン同様に、iOSの標準UIは存在しません。

ではどう表現するのかですが、ラジオボタンのUIと同様のUIが、チェックボックスに該当するUIとして使用されています。

同じく画像を用意することで、表現しているアプリも多く存在します。

その場合、アクティブと非アクティブの画像を用意します。

OSの話ではなく、デザインの意味で考えるのであれば、排他制御（項目内のどれか一つのみが選択されること。Document1を選択するとDocument2は選択されない状態）にはラジオボタンを使用し、複数選択可能な場合にチェックボックスを使用します。ユーザーに親切なのは、画像を用意してでもラジオボタンとチェックボックスを分けるデザインです。

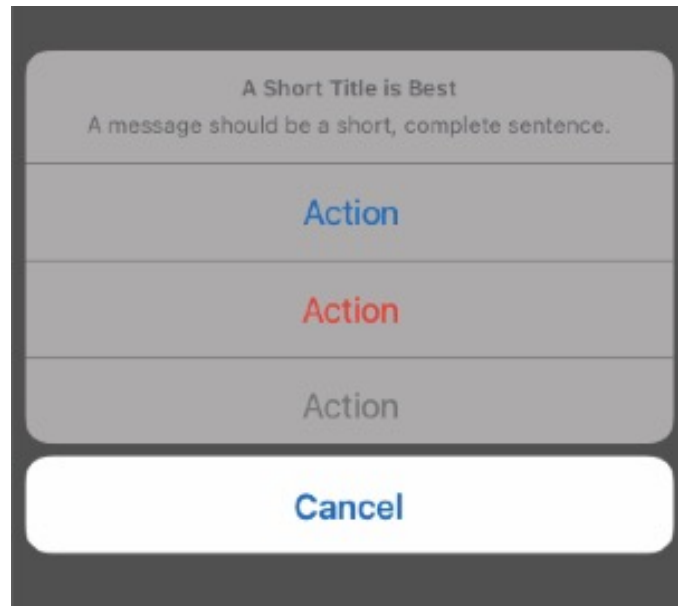




### 4-3-9. アクションシートとダイアログ

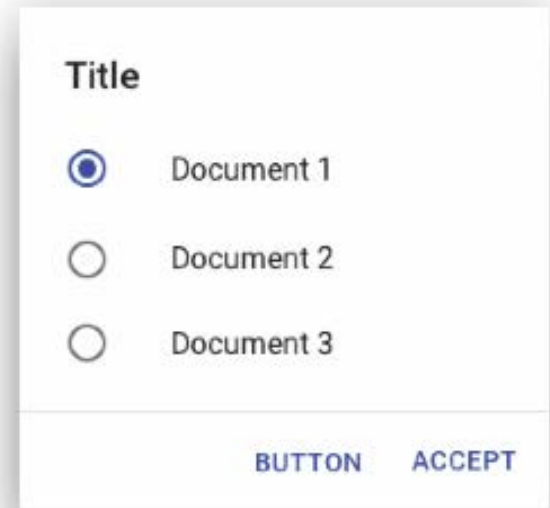
ユーザーへ、明瞭な行動を選択させる場合に使用する標準UIをiOSはアクションシートといい、アクションとキャンセルボタンがセットになっています。

画面下部に補助的なメニューをモーダル表示するものです。メニューの項目をどれかを選択する（タップする）するか操作をキャンセルすることで、シートを閉じることが可能です。



Androidの場合、明確な行動を選択させる場合には、ダイアログを使用します。同じユーザー行動を促すことが可能なUI選定を意識することが大切です。

※ダイアログは例を示しているだけで、複数選択のチェックボタンがあったり、ラジオボタンもない項目表示だけのものも存在します。



#### 4-3-10. リスト

iOS のリストビューには右端に矢印アイコンが表示されますが、Android のリストビューには基本的に矢印がつきません。また、細かいですが iOS はインデントのように線の左端に余白があるのですが、Android は端まで線が通ります。

iOS と Android を同じ UI にするというクライアントからのオーダーがあり、iOS に寄せてあるアプリは多く存在しています。

ターゲットが押せる感がわからないというレビューがあるという理由から矢印をつける場合もあります。

実際の「押せる感」問題は、実在のターゲットになり得るユーザーに触ってもらい確認してみないことには検証はできません。別の理由がある可能性もあります。

クライアントやUXデザイナーに協力いただき、実際の調査が可能かを提案できるならば視野に入れておきましょう。



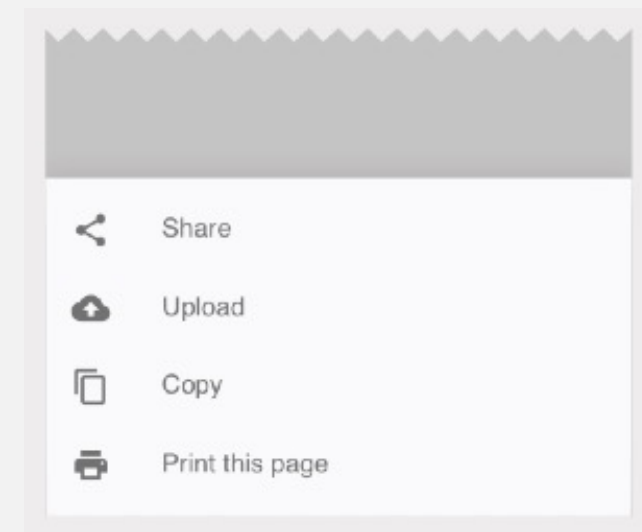
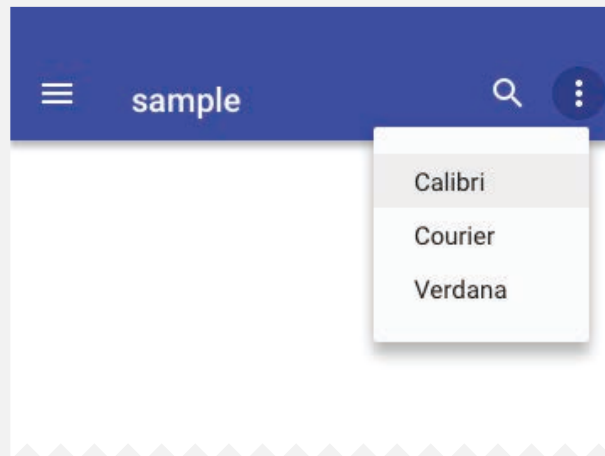
### 4-3-11. オプションメニュー

Androidでオプションメニューには、現在のアクティビティコンテキストに関連するアクションとその他のオプションを配置します。

iOSの標準コンポーネントには存在しないため、アクションシートやActivity Viewなどで代替される場合もあります。

デザインを行う上では、ガイドラインや、違いを認識した上で、ユーザーにとって何が最適なのか（迷わずに使えるのか）を選定していく作業がデザインの大切な作業です。

Android オプションメニュー



## 解答

問11 以下の、空欄を埋めなさい。

スマートフォンアプリの開発において、代表的なデザインガイドラインがあります。  
iOSは  、Androidは  が存在します。

### 【解答】

① Human Interface Guidelines    ② Material Design

---

## 解答

問12 スマートフォンの解像度について、正しい記述を選択しなさい。

- ア. スマートフォンの解像度は1インチの正方形の中にどれだけドットが存在するか「密度」で表します。
- イ. スマートフォンの解像度は、画面上にあるピクセルの数量で表します。

**【解答】 イ**

---

●密度で表示するのは、印刷の解像度です。

## 解答

問13 iOSの素材を書き出す場合の形式を、すべて選択してください。

ア. ai    イ. psd    ウ. pdf    エ. png    オ. svg    カ. jpg  
キ. Gif    ク. JSON    ケ. webp

**【解答】** ウ、エ、オ、カ、キ、ク

---

ア・イ： Adobeのファイル形式

ク： Lottieアニメーションなどを作成したときに使用することもあります。

ケ： Googleの画像フォーマット。

## 解答

問14 Andoroid 素材を書き出す場合に必要な倍数を、以下より選択してください。

ア. 0.75 倍    イ. 1.0 倍    ウ. 1.5 倍    エ. 2.0 倍    オ. 3.0 倍  
カ. 4.0 倍    キ. 5.0 倍    ク. 6.0 倍

**【解答】** イ、ウ、エ、オ、カ

---

## 解答

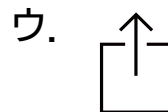
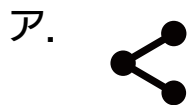
問15 Andoroid のバックボタンの挙動を記述してください。

「戻る (バック) ボタン」はアプリを超えて「ひとつ前のアクティビティに移動する」ものです。



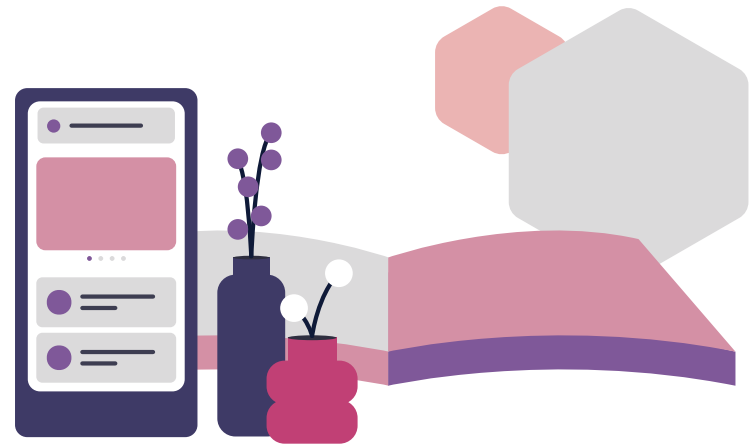
## 解答

問16 iOSのシェアアイコンを選択してください。



**【解答】** イ、ウ

- アはAndroidのシェアです。近年は、Youtubeなどの広がりにより共有の認識が広がっています。共有認識は時代ともに変化しますので、キャッチアップしていきましょう。



# Appendix

付録

## おすすめ書籍

アシスタントデザイナーの方に、おすすめ書籍を追記したほうがよいというフィードバックをいただいたので記載します。



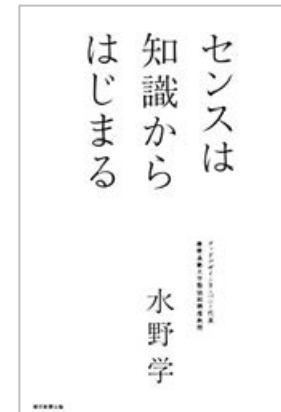
- 基本はかんたん配色のルール デザイン ビギナー シリーズ



- エモーショナルデザインの実践 — 感性ともものをつなぐプロダクトデザインの考えかた



- オブジェクト指向 UIデザイン — 使いやすいソフトウェアの原理



- センスは知識からはじまる

## おすすめ書籍

アシスタントデザイナーの方に、おすすめ書籍を追記したほうがよいというフィードバックをいただいたので記載します。



- ファスト&スロー  
あなたの意思はどの  
ように決まる  
か?



- 融けるデザイン  
ハード×ソフト×  
ネット時代の新た  
な設計論



- 観察力の鍛え方  
一流のクリエイ  
ターは世界をどう  
見ているのか



- 考えなしの行動?  
(左の解説本の原  
書です)

## おすすめ書籍

アシスタントデザイナーの方に、おすすめ書籍を追記したほうがよいというフィードバックをいただいたので記載します。



- クリエイティブ・マインドセット  
想像力・好奇心・勇気が目覚める驚異の思考法



- 予想どおりに不合理: 行動経済学が明かす「あなたがそれを選ぶわけ」

デザインの具体的なやり方などが書かれた本については、あまりご紹介していません。初心者の方は、WEBサービスなどを利用されるとよいでしょう。

- [Daily UI](#)
- [Collect UI](#)
- [Cocoda Training](#)

## おわりに

このドリルは、未完成です。

まだ記載できてないことも多くあり、弊社クラスメソッドのデザイン組織が作られていく中で変わっていくものもあります。「ver.1.0」とバージョンをつけたのは、はじまりと更新する意志の表明です。

今これを読んでいる方が、更新をする未来もあるでしょう。

基礎の話ばかり書きましたが、弊社の経営理念「オープンな発想と高い技術力により、すべての人々の創造活動に貢献し続ける。」の通りに、デザイナーのオープンな発想を止めるものではありません。むしろ、基礎を知った上での「発想」は求められると思います。

その上で、自分のなかでの「基礎、デザイン信念」といった骨格を完成させていく手助けの一歩として使用していただければ幸いです。

## 感謝

今回のドリル作成に際し、株式会社ページの「WEBデザインドリル」を参考とさせていただきました。弊社CDOでもある粉谷より資料提供いただいたのですが、大変ありがとうございました。

壁打ちに協力いただいたモバイル事業部PdM、友人デザイナーのみなさま。

内容についてのレビューに付き合ってくださいましたモバイル事業部エンジニアの皆様。LGTMをありがとう。

すてきなイラストを書いてくださった弊社デザイナーさま。

他に仕事がある中、確認いただいた弊社デザイナーのみなさま。

心より感謝申し上げます。

2022/01/31

クラスメソッド株式会社 デザイナー 田中由希子

## 変更履歴

- 2022/02/09 ver1.1 軽微な文字修正を行いました。
- 2022/01/31 ver1.0





**classmethod**