now loading

blanktar.jp

目黒研究室 基礎技術講座

blanktar.jp

講座の流れ

- 11月 2日 **プロトタイピングを支える技術**
  卒研で使うかもしれない技術の紹介。

- 11月16日 **デザインからプロトタイピングまで**
  要件を決めて、プロトタイプを作って動かすまでの流れ。

- 12月14日 **最低限の統計学**
  教授にボコボコにされないためのデータの集め方、使い方。

講座の流れ

- 11月 2日 **プロトタイピングを支える技術**
  卒研で使うかもしれない技術の紹介。

- 11月16日 **デザインからプロトタイピングまで**
  要件を決めて、プロトタイプを作って動かすまでの流れ。

- 12月14日 **最低限の統計学**
  教授にボコボコにされないためのデータの集め方、使い方。

この講座について

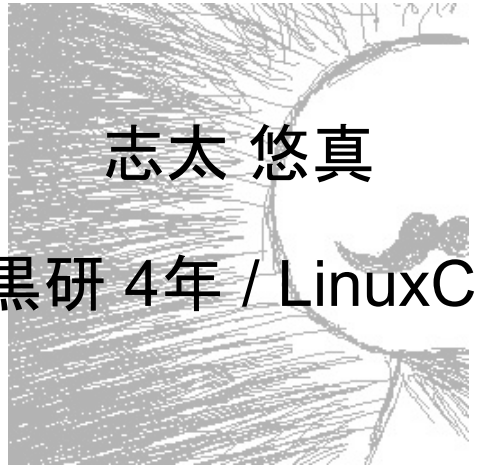
卒業研究で役に立ちそうな知識をお届けする講座。全3回。

ぶっちゃけ時間が無いのでざっくりやります。
詳しく知りたい方は直接聞いてください。

スライドは補足資料付きで公開してあります。

goo.gl/h8Kp7T

講師について



志太 悠真

目黒研 4年 / LinuxClub

m@crat.jp

@macrat_jp

目黒研究室 基礎技術講座 第二回

デザインからプロトタイピングまで

blanktar.jp

前回の反省

前回の反省

退屈だった

というわけで

TL;DR.

最初の5分で言いたいこと全部言います。

TL;DR.

あとは寝てても良いよ。

TL;DR.

~~あとは寝てても良いよ。~~

Too Long; Don't Read.

どんな流れで開発するか？

一般論: ウォーターフォールモデル

ウォーターフォールモデル

要件定義
↓
概要設計
↓
詳細設計
↓
開発
↓
テスト

こんなやつ

ウォーターフォールモデル

要件定義

↓

概要設計

↓

詳細設計

↓

開発

↓

テスト

何を作るのか

を決める

ウォーターフォールモデル

要件定義

↓

概要設計

↓

詳細設計

↓

開発

↓

テスト

何で作るか

を決める

ウォーターフォールモデル

要件定義

↓

概要設計

↓

詳細設計

↓

開発

↓

テスト

どう作るか

を決める

ウォーターフォールモデル

要件定義

↓

概要設計

↓

詳細設計

↓

開発

↓

テスト

がんばって作る

ウォーターフォールモデル

要件定義

↓

概要設計

↓

詳細設計

↓

開発

↓

テスト

動くか確認

ウォーターフォールモデル

要件定義
↓
概要設計
↓
詳細設計
↓
開発
↓
テスト

一般的な大規模開発では
この流れが多い

ウォーターフォールモデル

要件定義
↓
概要設計
↓
詳細設計
↓
開発
↓
テスト

一般的な**大規模開発**では
この流れが多い

ウォーターフォールモデル

要件定義
↓
概要設計
↓
詳細設計
↓
開発
↓
テスト

ウォーターフォールの特徴

手戻りに弱い

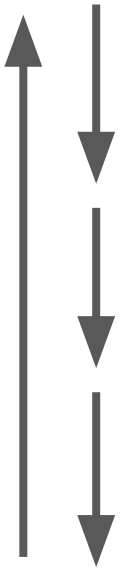
ウォーターフォールモデル

要件定義
↓
概要設計
↓
詳細設計
↓
開発
↓
テスト

一度でも
戻ってしまうと

ウォーターフォールモデル

要件定義
↓
概要設計
↓
詳細設計
↓
開発
↓
テスト



全部
やり直しになる

ウォーターフォールモデル

要件定義
↓
概要設計
↓
詳細設計
↓
開発
↓
テスト

多分やめた方が良い

ではどうするか

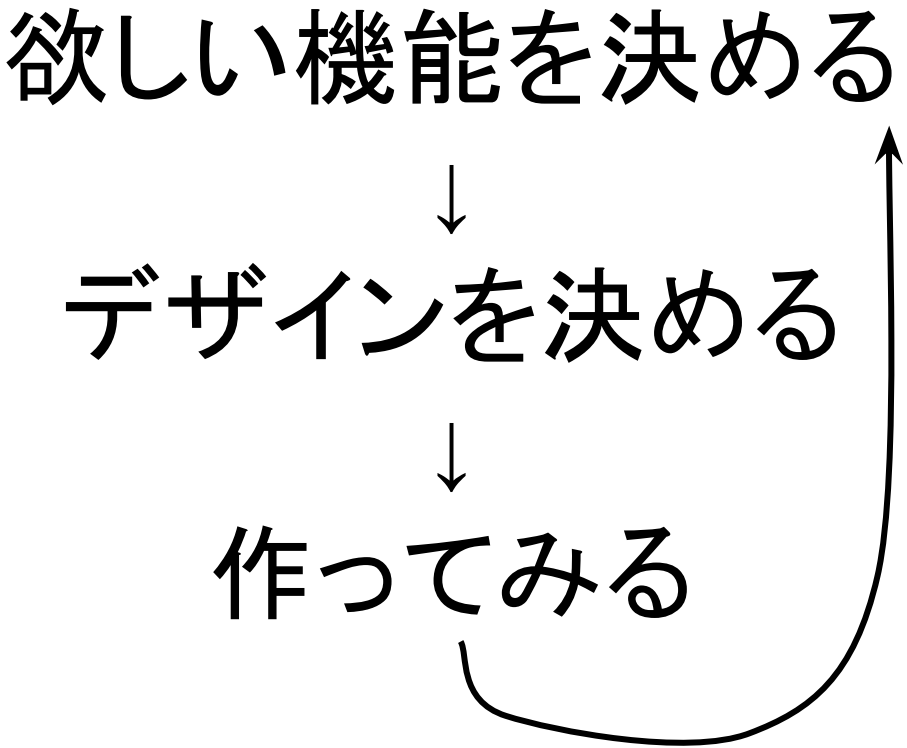
欲しい機能を決める

↓

デザインを決める

↓

作ってみる

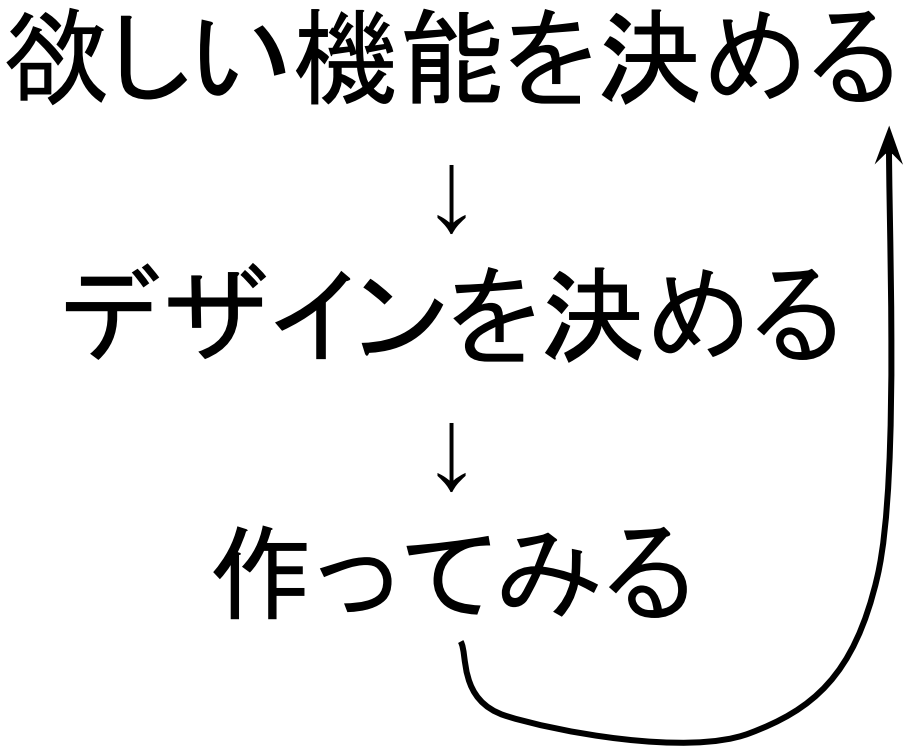

欲しい機能を決める
↓
デザインを決める
↓
作ってみる

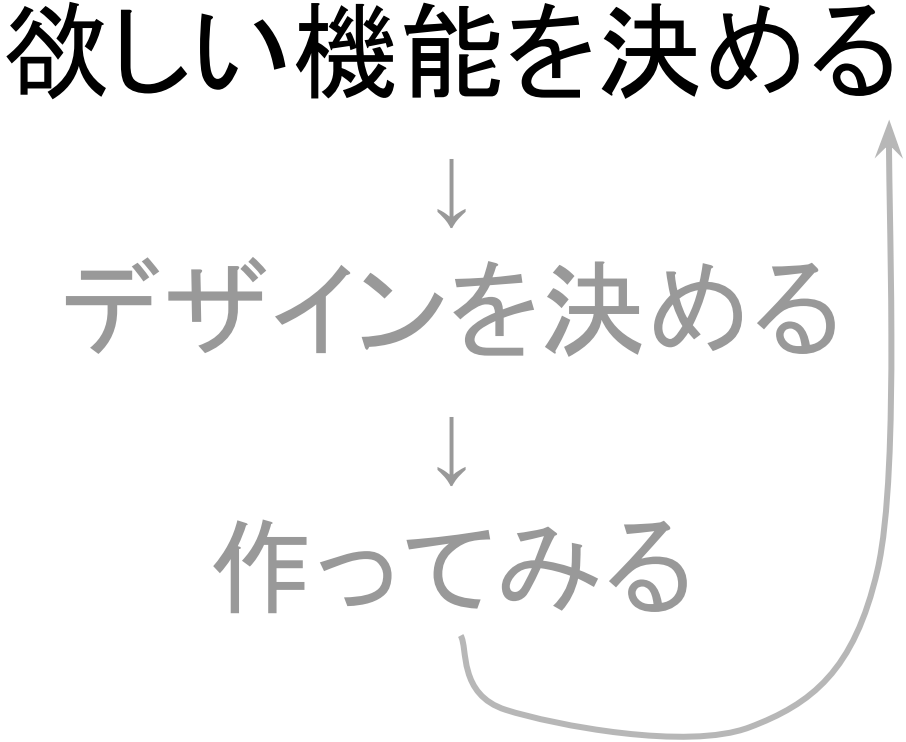
例えば:

短文でやりとり出来るSNS作りたい

例えば:

短文でやりとり出来るSNS作りたい


欲しい機能を決める
↓
デザインを決める
↓
作ってみる


欲しい機能を決める
↓
デザインを決める
↓
作ってみる

- アカウントがある
- 呟ける
- 人をフォローできる

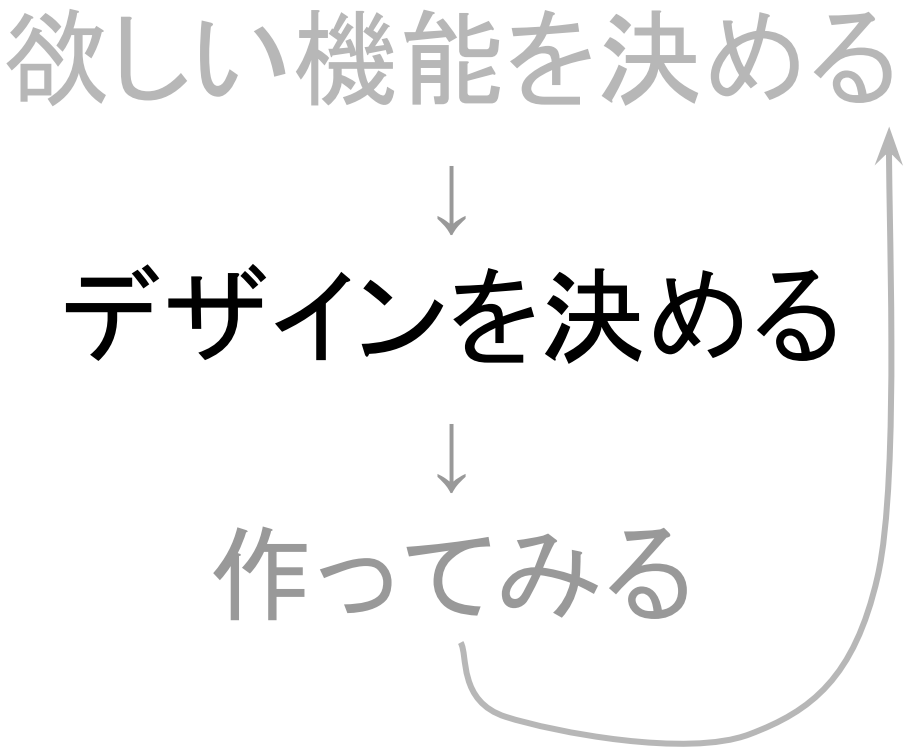
- アカウントがある
- 呟ける
- 人をブロックできる

注意

最初から全部作ろうとしてはいけない

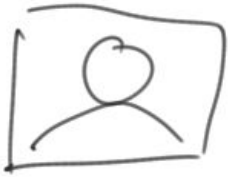
- アカウントがある
- 呟ける
- 人をフォローできる

目標：アカウント機能を作る


欲しい機能を決める
↓
デザインを決める
↓
作ってみる

タイトルバー

ログイン



名前

プロフィールとか

名前

つぶやき

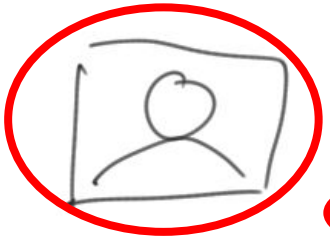
名前

つぶやき

名前

タイトルバー

ログイン



名前

プロフィールとか〜

名前

つぶやき〜

名前

つぶやき〜

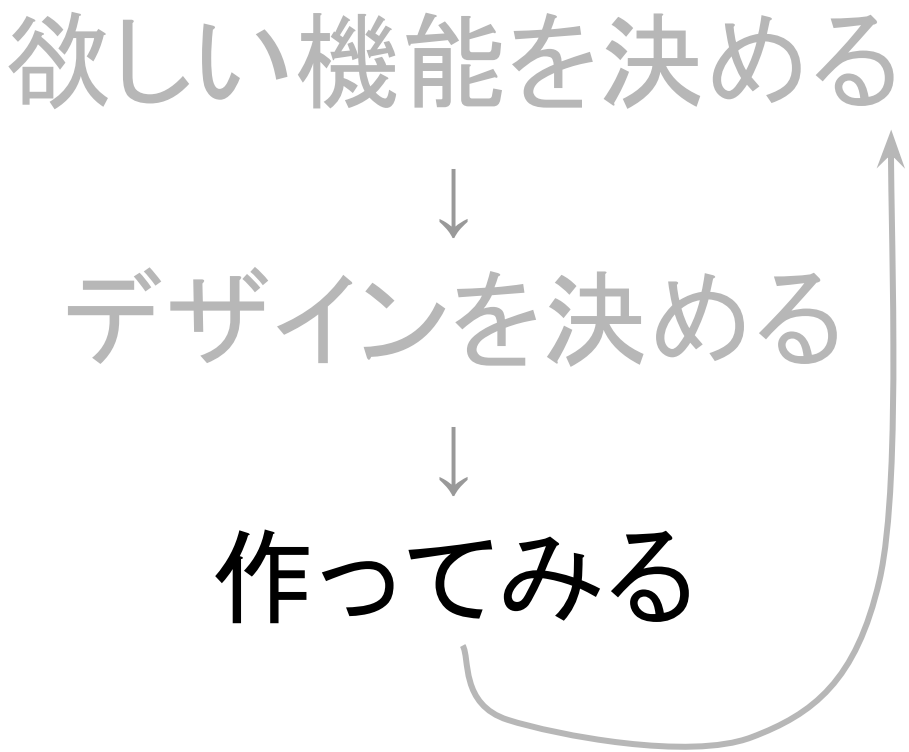
名前

ログイン/ログアウト

名前（ID）

アイコン

プロフィール


欲しい機能を決める
↓
デザインを決める
↓
作ってみる

外側を作ってから

中身を作り込むと良い

かもしれない

目黒研 技術講座 - Go
第二回 デザインから
profile.html
MacRat
file:///mnt/tmpfs/profile.html

タイトルバー　ログイン

名前

プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

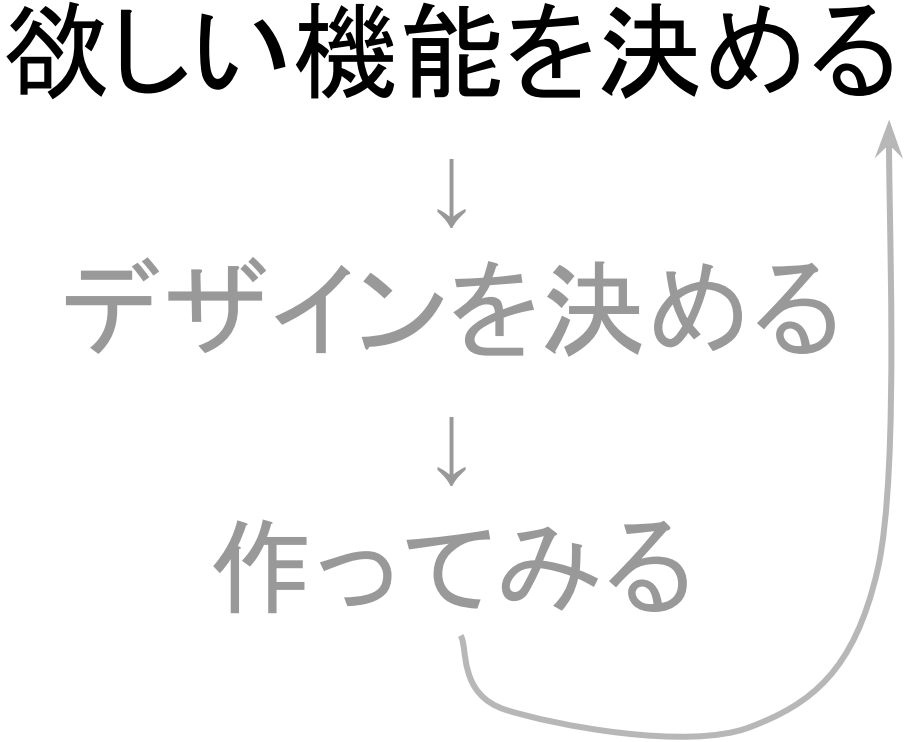
名前

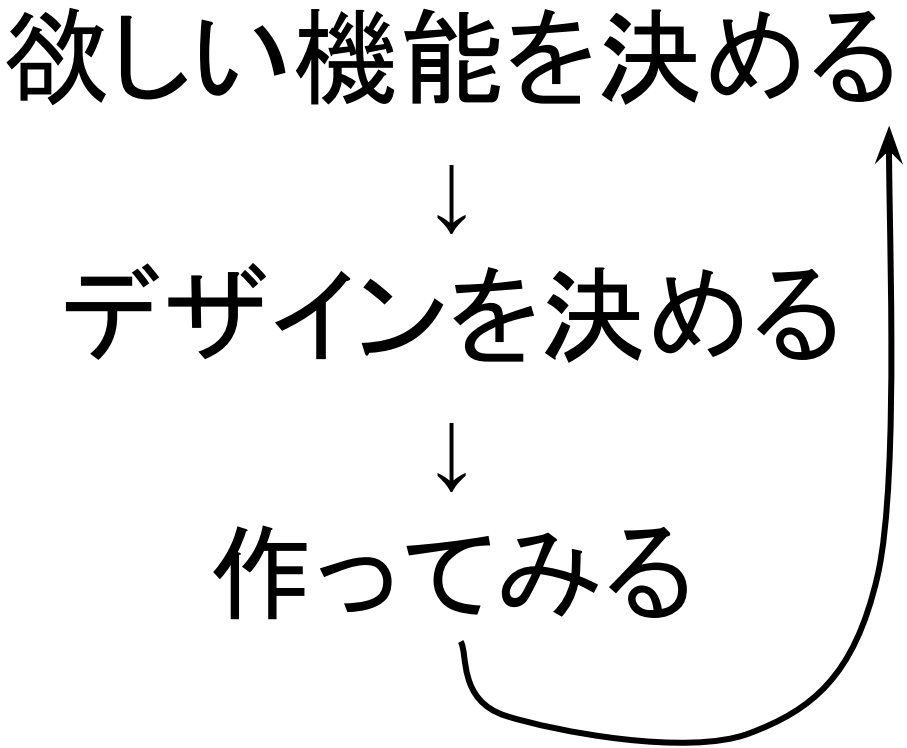
つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

目標：アカウント機能を作る

目標：アカウント機能を作る

達成！


欲しい機能を決める
↓
デザインを決める
↓
作ってみる


欲しい機能を決める
↓
デザインを決める
↓
作ってみる

小さな機能ごとに作ると楽

これをプロトタイピングモデルとか言う

（厳密にはちょっと違う）

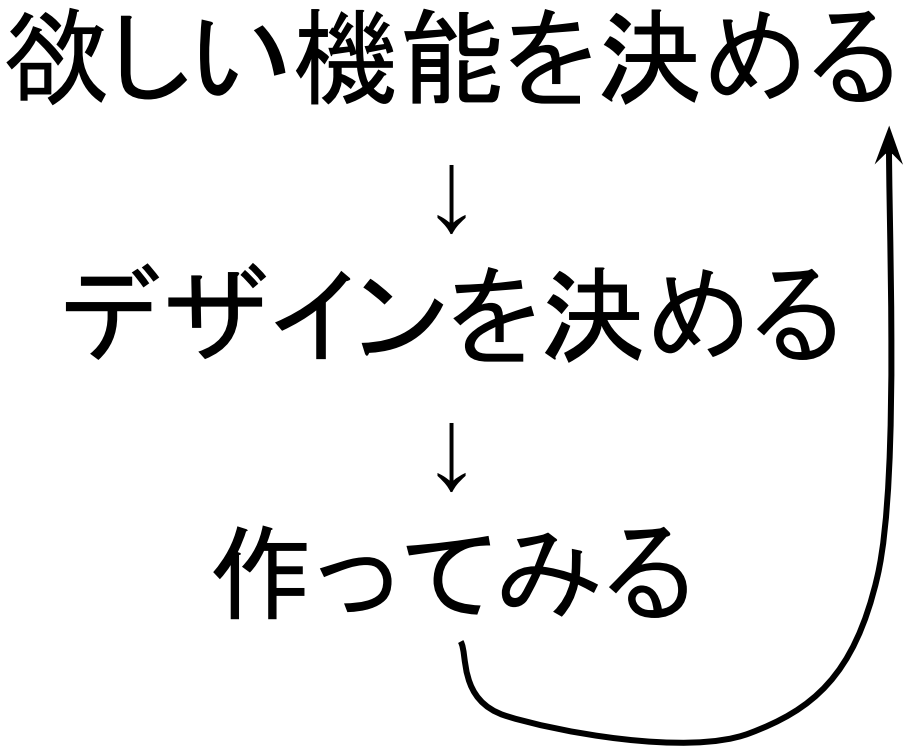
終

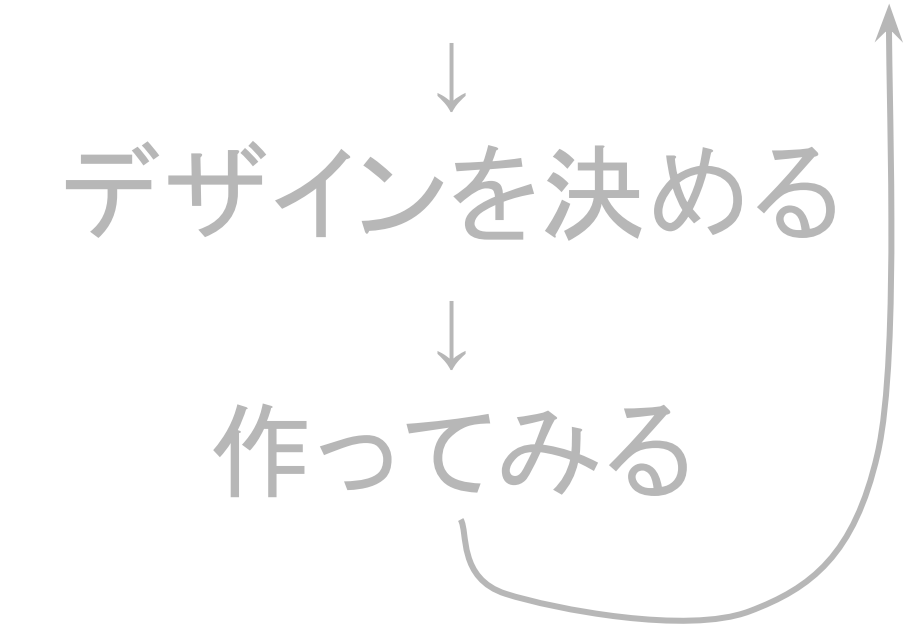
前半戦終了

ここから詳細に入ります

前半戦終了

おやすみなさい


欲しい機能を決める
↓
デザインを決める
↓
作ってみる


欲しい機能を決める
デザインを決める
作ってみる

さっきの例だと

短文でやりとり出来るSNS作りたい

掘り下げると

- アカウントがある
- 呟ける
- 人をフォローできる

掘り下げると

- アカウントがある
- 呟ける
- 人をフォローできる

もっと掘り下げると

- ログイン/ログアウト
- 名前（ID）
- アイコン
- プロフィール

もっと掘り下げると

- ログイン/ログアウト
- 名前（ID）
- アイコン
- プロフィール

もっともっと掘り下げると

- 表示
- 編集
- 非公開設定

掘り下げ続けると

キリが無い

なので、適当なところで
画面のデザインをする

プロフィール画面が欲しい

くらいの粒度がちょうど良いと思う

- アカウントがある
- 呟ける
- 人をフォローできる

だいたいこのぐらい

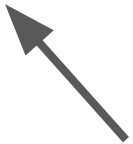
- プロフィール画面
- 投稿画面
- タイムライン画面

言い変えるとこんな感じ

どれを最初に作るか

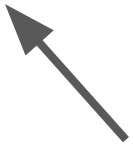
- プロフィール画面
- 投稿画面
- タイムライン画面

- プロフィール画面
- 投稿画面
- タイムライン画面



投稿が無いと作れない

- プロフィール画面
- 投稿画面
- タイムライン画面



他の機能に
依存してる物は後回し

- プロフィール画面
- 投稿画面
- タイムライン画面

- プロフィール画面
- 投稿画面
- タイムライン画面

どっちでも良い

と思う

無いと話にならない物が最優先

あると楽しいものが優先

面倒臭そうなものも優先

無いと話にならない物が最優先

あると楽しいものが優先

面倒臭そうなものも優先

無いと話にならない物が最優先

あると楽しいものが優先

面倒臭そうなものも優先

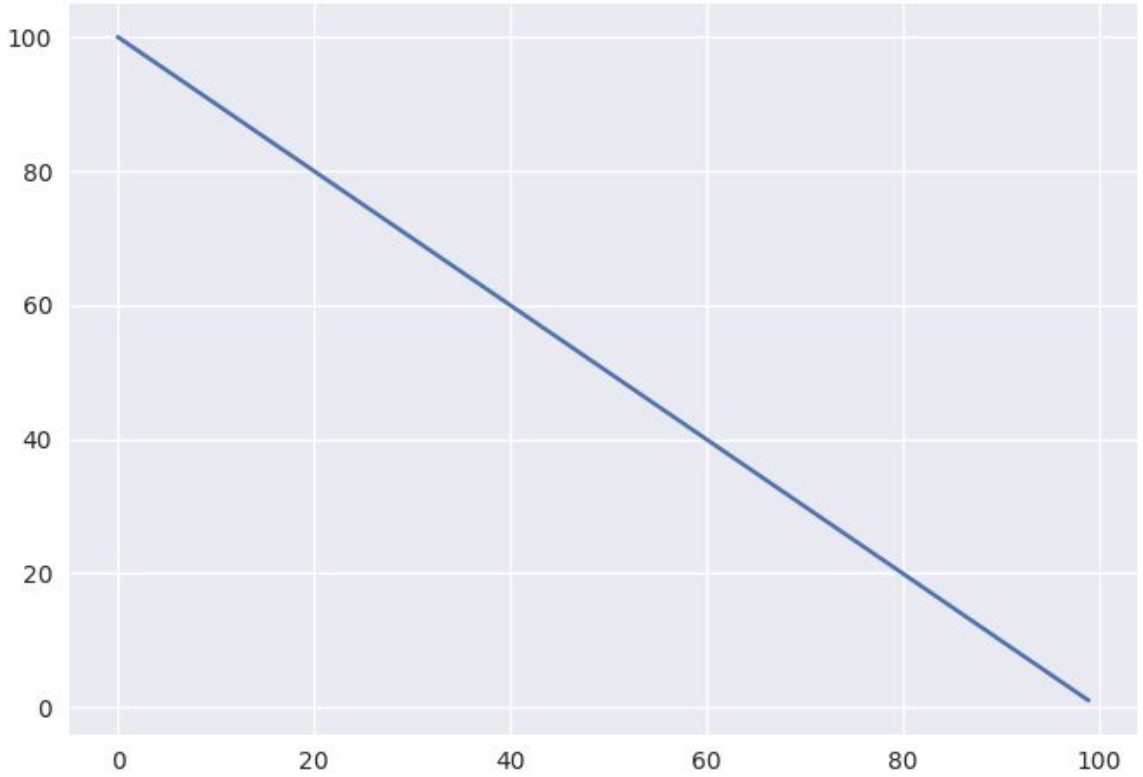
無いと話にならない物が最優先

あると楽しいものが優先

面倒臭そうなものも優先

モチベーションの曲線

モチベーションの曲線



※画像はイメージです。感想には個人差があります。

見た目が出来れば

ちょっとたのしい

たのしければ

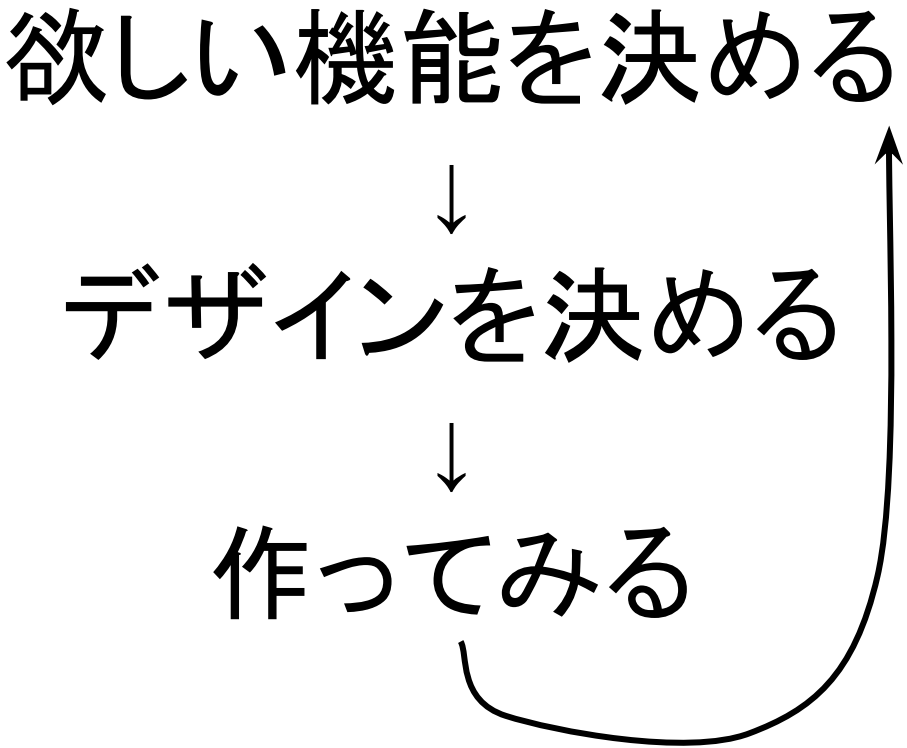
ちょっと頑張れる

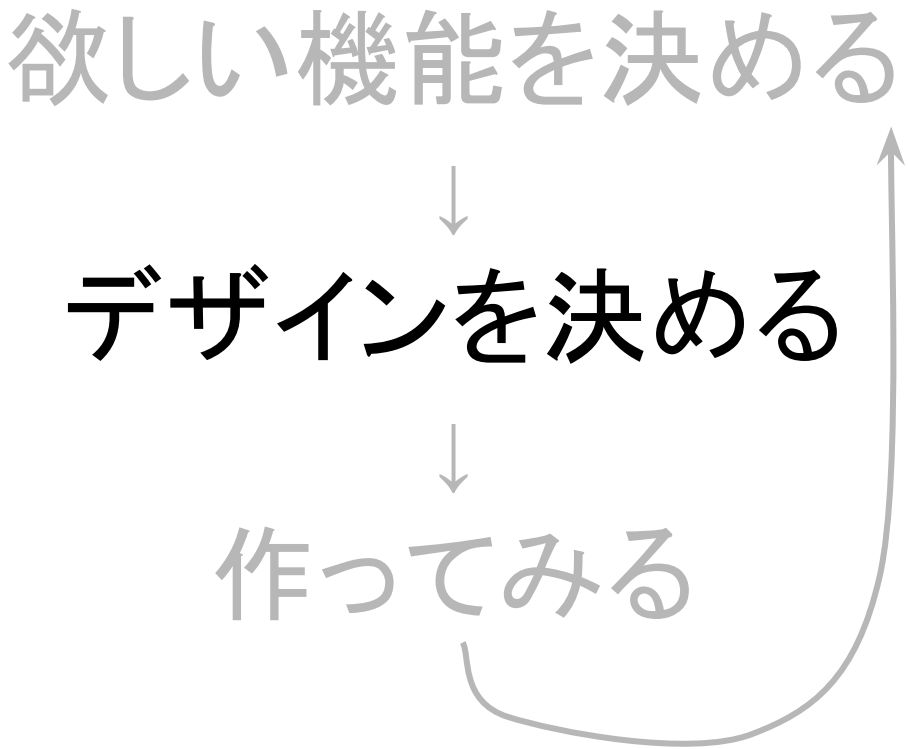
面倒なものは先に

楽しいものも先に

瑣末なことは後で

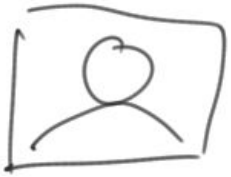
モチベーション = 100% - 時間 * 1/楽しさ


欲しい機能を決める
↓
デザインを決める
↓
作ってみる


欲しい機能を決める
デザインを決める
作ってみる

タイトルバー

ログイン



名前

プロフィールとか

名前

つぶやき

名前

つぶやき

名前

- 必要な機能が分かりやすくなる
- モチベーションが上がる

- 必要な機能が分かりやすくなる
- モチベーションが上がる
- ゼミでの進捗発表が楽

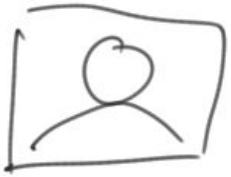
デザインの作り方

- お絵描きする
- ツールを使う
- HTML書いちゃう

- お絵描きする
- ツールを使う
- HTML書いちゃう

タイトルバー

ログイン



名前

プロフィールとか

名前

つぶやき

名前

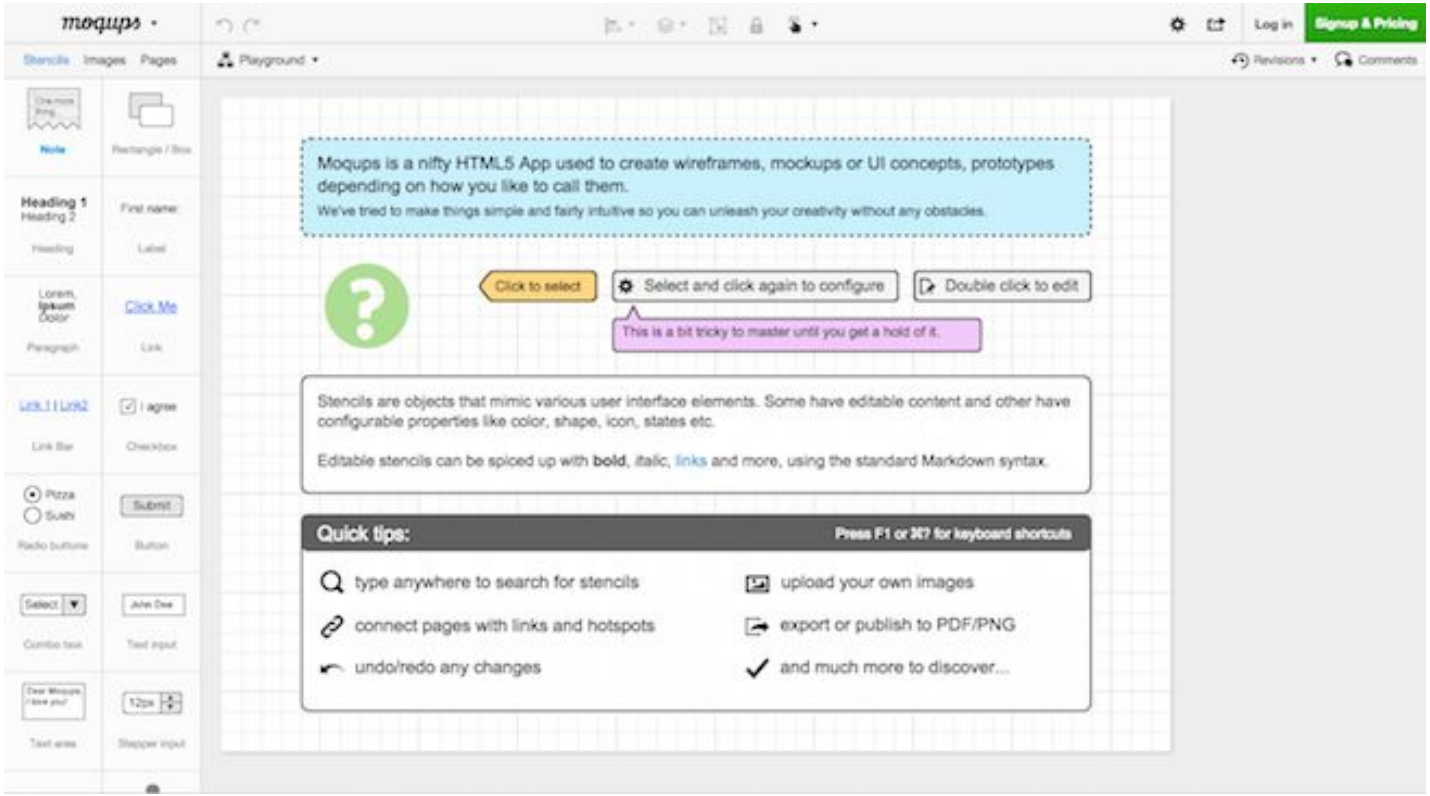
つぶやき

名前

POPとかいうやつ

- お絵描きする
- ツールを使う
- HTML書いちゃう

moqupsとかいうやつ



https://moqups.com/

ワイヤーフレームツール

とかググると色々出てくる

- お絵描きする
- ツールを使う
- HTML書いちゃう

目黒研 技術講座 - Go
第二回 デザインから
profile.html
MacRat
file:///mnt/tmpfs/profile.html

タイトルバー　ログイン

名前

プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

名前

つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

名前

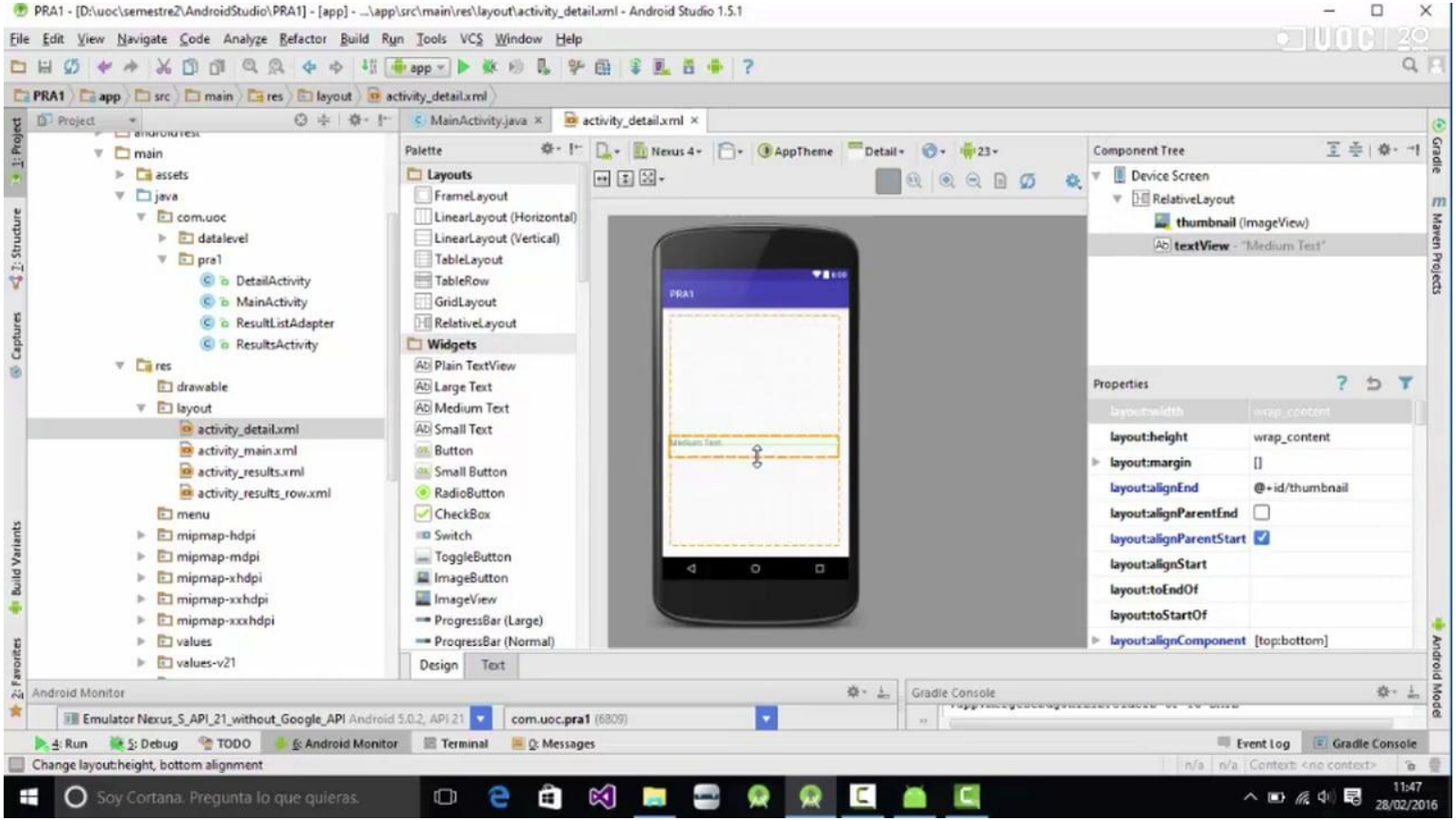
つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき

```html
<style>
	header { border-bottom: 2px solid gray; width: 100%; padding: .1em; }
	header > a { float: right; padding-right: .1em; }
	main { width: 80%; margin: .5em 10% 0; border: 0 solid gray; border-width: 0 2px; min-height: 20%; }
	#profile { padding: 0 .5em.5em; border-bottom: 2px solid gray; display: flex; }
	#profile > img { width: 7em; height: 7em; flex-shrink: 0; }
	#profile > div { padding: 1em; }
	h1 { margin: 0; }
	#tweets > div { display: flex; margin: 0 .5em; padding: 1em 0; border-bottom: 1px solid gray; }
	#tweets img { height: 3em; width: 3em; margin-right: 1em; }
	h2 { margin: 0 0 .4em; font-size: 1rem; }
</style>
<header>タイトルバー <a href>ログイン</a></header>
<main>
	<div id=profile>
		<img src="icon.png"><div><h1>名前</h1> プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール</div>
	</div>
	<div id=tweets>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
	</div>
</main>
```

- お絵描きする
- ツールを使う
- HTML書いちゃう

アプリの場合

IDE付属のデザイナー

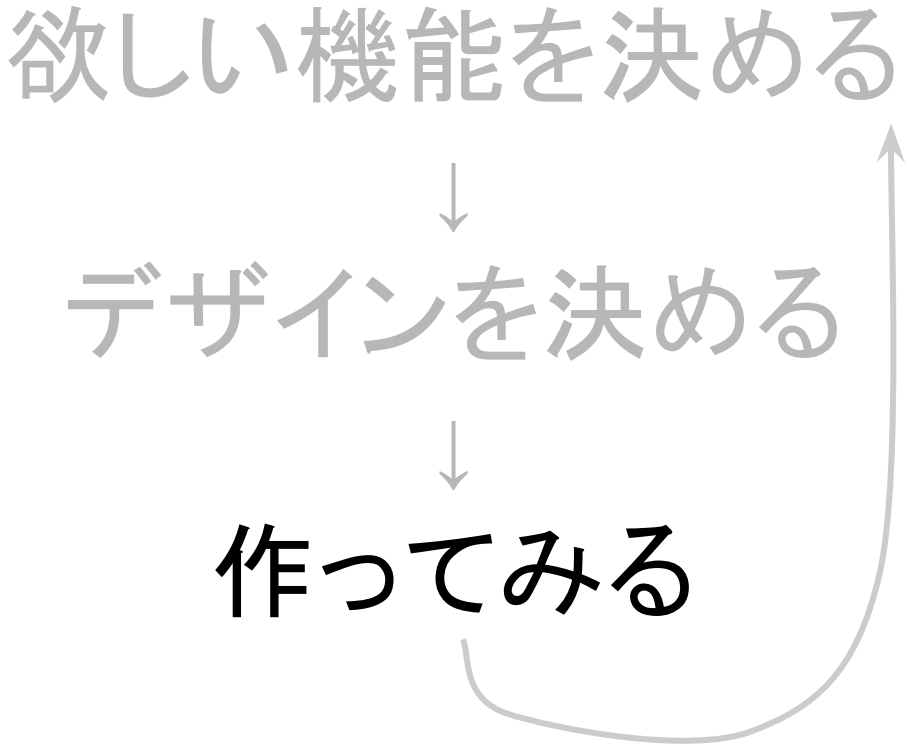


IDE付属のデザインツール

そのまま開発に使える

見た目がリアル(というかそのもの)

個人的にはお絵描きがしっくり来る


欲しい機能を決める
↓
デザインを決める
↓
作ってみる

ここでも見た目から作る

と、良いと思う

```html
<style>
	header { border-bottom: 2px solid gray; width: 100%; padding: .1em; }
	header > a { float: right; padding-right: .1em; }
	main { width: 80%; margin: .5em 10% 0; border: 0 solid gray; border-width: 0 2px; min-height: 20%; }
	#profile { padding: 0 .5em.5em; border-bottom: 2px solid gray; display: flex; }
	#profile > img { width: 7em; height: 7em; flex-shrink: 0; }
	#profile > div { padding: 1em; }
	h1 { margin: 0; }
	#tweets > div { display: flex; margin: 0 .5em; padding: 1em 0; border-bottom: 1px solid gray; }
	#tweets img { height: 3em; width: 3em; margin-right: 1em; }
	h2 { margin: 0 0 .4em; font-size: 1rem; }
</style>
<header>タイトルバー <a href>ログイン</a></header>
<main>
	<div id=profile>
		<img src="icon.png"><div><h1>名前</h1> プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール</div>
	</div>
	<div id=tweets>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
	</div>
</main>
```

```
<style>
	header { border-bottom: 2px solid gray; width: 100%; padding: .1em; }
	header > a { float: right; padding-right: .1em; }
	main { width: 80%; margin: .5em 10% 0; border: 0 solid gray; border-width: 0 2px; min-height: 20%; }
	#profile { padding: 0 .5em.5em; border-bottom: 2px solid gray; display: flex; }
	#profile > img { width: 7em; height: 7em; flex-shrink: 0; }
	#profile > div { padding: 1em; }
	h1 { margin: 0; }
	#tweets > div { display: flex; margin: 0 .5em; padding: 1em 0; border-bottom: 1px solid gray; }
	#tweets img { height: 3em; width: 3em; margin-right: 1em; }
	h2 { margin: 0 0 .4em; font-size: 1rem; }
</style>
<header>タイトルバー <a href>ログイン</a></header>
<main>
	<div id=profile>
		<img src="icon.png"><div><h1>名前</h1> プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール</div>
	</div>
	<div id=tweets>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
	</div>
</main>
```

ユーザー名

```html
<style>
	header { border-bottom: 2px solid gray; width: 100%; padding: .1em; }
	header > a { float: right; padding-right: .1em; }
	main { width: 80%; margin: .5em 10% 0; border: 0 solid gray; border-width: 0 2px; min-height: 20%; }
	#profile { padding: 0 .5em.5em; border-bottom: 2px solid gray; display: flex; }
	#profile > img { width: 7em; height: 7em; flex-shrink: 0; }
	#profile > div { padding: 1em; }
	h1 { margin: 0; }
	#tweets > div { display: flex; margin: 0 .5em; padding: 1em 0; border-bottom: 1px solid gray; }
	#tweets img { height: 3em; width: 3em; margin-right: 1em; }
	h2 { margin: 0 0 .4em; font-size: 1rem; }
</style>
<header>タイトルバー <a href>ログイン</a></header>
<main>
	<div id=profile>
		<img src="icon.png"><div><h1>名前</h1> プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール</div>
	</div>
	<div id=tweets>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
	</div>
</main>
```

ユーザー名

プロフィール

```html
<style>
	header { border-bottom: 2px solid gray; width: 100%; padding: .1em; }
	header > a { float: right; padding-right: .1em; }
	main { width: 80%; margin: .5em 10% 0; border: 0 solid gray; border-width: 0 2px; min-height: 20%; }
	#profile { padding: 0 .5em.5em; border-bottom: 2px solid gray; display: flex; }
	#profile > img { width: 7em; height: 7em; flex-shrink: 0; }
	#profile > div { padding: 1em; }
	h1 { margin: 0; }
	#tweets > div { display: flex; margin: 0 .5em; padding: 1em 0; border-bottom: 1px solid gray; }
	#tweets img { height: 3em; width: 3em; margin-right: 1em; }
	h2 { margin: 0 0 .4em; font-size: 1rem; }
</style>
<header>タイトルバー <a href>ログイン</a></header>
<main>
	<div id=profile>
		<img src="icon.png"><div><h1>名前</h1> プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール</div>
	</div>
	<div id=tweets>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
		<div><img src="icon.png"><div><h2>名前</h2> つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき</div></div>
	</div>
</main>
```

ユーザーアイコン

ユーザー名

プロフィール

デザインを作って

穴埋めの感覚で作る

ほとんどの場合

テンプレートエンジン

というのを使うことになる

```
<header>タイトルバー <a href>ログイン</a></header>
<main>
	<div id=profile>
		<img src="icon.png"><div><h1>名前</h1> プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール プロフィール</div>
	</div>
	<div id=tweets>
		<div>
			<img src="icon.png">
			<div>
				<h2>名前</h2>
				つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき
			</div>
		</div>
		<div>
			<img src="icon.png">
			<div>
				<h2>名前</h2>
				つぶやき つぶやき つぶやき つぶやき つぶやき つぶやき
			</div>
		</div>
	</div>
</main>
```

```
<main>
  <div id=profile>
    <img src="{{ url_for('user_icon', id=user.id) }}"><div><h1>{{ user.name }}</h1>{{ user.profile }}</div>
  <div id=tweets>
    {% for tweet in user.tweets %}
      <div>
        <img src="{{ url_for('user_icon', id=tweet.user_id) }}">
        <div>
          <h2>{{ tweet.user_name }}</h2>
          {{ tweet.content }}
         </div>
      </div>
    {% endfor %}
  </div>
</main>
```

※これはjinja2というエンジンの書き方です。
書き方は物によって違います。

```
<main>
  <div id=profile>
    <img src="{{ url_for('user_icon', id=user.id) }}"><div><h1>{{ user.name }}</h1>{{ user.profile }}</div>
  <div id=tweets>
    {% for tweet in user.tweets %}
      <div>
        <img src="{{ url_for('user_icon', id=tweet.user_id) }}">
        <div>
          <h2>{{ tweet.user_name }}</h2>
          {{ tweet.content }}
         </div>
      </div>
    {% endfor %}
  </div>
</main>
```

```
<main>
  <div id=profile>
    <img src="{{ url_for('user_icon', id=user.id) }}"><div><h1>{{ user.name }}</h1>{{ user.profile }}</div>
  <div id=tweets>
    {% for tweet in user.tweets %}
      <div>
        <img src="{{ url_for('user_icon', id=tweet.user_id) }}">
        <div>
          <h2>{{ tweet.user_name }}</h2>
          {{ tweet.content }}
         </div>
      </div>
    {% endfor %}
  </div>
</main>
```

```
<main>
  <div id=profile>
    <img src="{{ url_for('user_icon', id=user.id) }}"><div><h1>{{ user.name }}</h1>{{ user.profile }}</div>
  <div id=tweets>
    {% for tweet in user.tweets %}
      <div>
        <img src="{{ url_for('user_icon', id=tweet.user_id) }}">
        <div>
          <h2>{{ tweet.user_name }}</h2>
          {{ tweet.content }}
         </div>
      </div>
    {% endfor %}
  </div>
</main>
```

この範囲を置き換えてね

という指示

```
<main>
  <div id=profile>
    <img src="{{ url_for('user_icon', id=user.id) }}"><div><h1>{{ user.name }}</h1>{{ user.profile }}</div>
  <div id=tweets>
    {% for tweet in user.tweets %}
      <div>
        <img src="{{ url_for('user_icon', id=tweet.user_id) }}">
        <div>
          <h2>{{ tweet.user_name }}</h2>
          {{ tweet.content }}
         </div>
      </div>
    {% endfor %}
  </div>
</main>
```

普通にfor文とかもある

```
<main>
  <div id=profile>
    <img src="{{ url_for('user_icon', id=user.id) }}"><div><h1>{{ user.name }}</h1>{{ user.profile }}</div>
  <div id=tweets>
    {% for tweet in user.tweets %}
      <div>
        <img src="{{ url_for('user_icon', id=tweet.user_id) }}">
        <div>
          <h2>{{ tweet.user_name }}</h2>
          {{ tweet.content }}
         </div>
      </div>
    {% endfor %}
  </div>
</main>
```

穴埋め問題っぽい感覚

```
<main>
  <div id=profile>
    <img src="{{ ユーザーのアイコン }}"><div><h1>{{ ユーザー名 }}</h1>{{ ユーザープロフィール }}</div>
  <div id=tweets>
    {% ユーザーの全ての投稿にくり返し %}
      <div>
        <img src="{{ 投稿者のアイコン }}">
        <div>
          <h2>{{ 投稿者の名前 }}</h2>
          {{ 投稿の内容}}
        </div>
      </div>
    {% くり返し終わり %}
  </div>
</main>
```

穴埋め問題っぽい感覚

見た目から作ると

抜け・漏れが少ない

アプリの場合

テンプレートエンジン的なものは無い

なので、穴埋めには出来ない

デザイン

```
<LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">

        <ImageView
                android:id="@+id/user_icon"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />

        <Label
                android:id="@+id/profile"
                android:layout_width="match_parent"
                android:layout_height="match_parent" />
</LinearLayout>
```

プログラム

```
class MainActivity : Activity {
  override fun onCreate(savedInstanceState: Bundle) {
    findViewById(R.id.user_icon).imageResource
        = user_icon_resource;

    findViewById(R.id.profile).imageResource
        = profile_text;
  }
}
```

※Android / kotlinの場合

デザイン

```
<LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">

        <ImageView
                android:id="@+id/user_icon"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />

        <Label
                android:id="@+id/profile"
                android:layout_width="match_parent"
                android:layout_height="match_parent" />
</LinearLayout>
```

プログラム

```
class MainActivity : Activity {
  override fun onCreate(savedInstanceState: Bundle) {
    findViewById(R.id.user_icon).imageResource
        = user_icon_resource;

    findViewById(R.id.profile).imageResource
        = profile_text;
  }
}
```

※Android / kotlinの場合

HTMLの場合は穴埋めっぽい感じ

アプリの場合は紐付ける感じ

見た目から作ると

抜け・漏れが少ない

「あれ、このボタン反応しないけど」

注意

途中から言語やライブラリを変えるのは面倒臭い

注意

かなり面倒臭い

作り始める前によく調べる

試しに簡単なものを作ってみたりする

ある程度作ったらあとはなるべく変えない

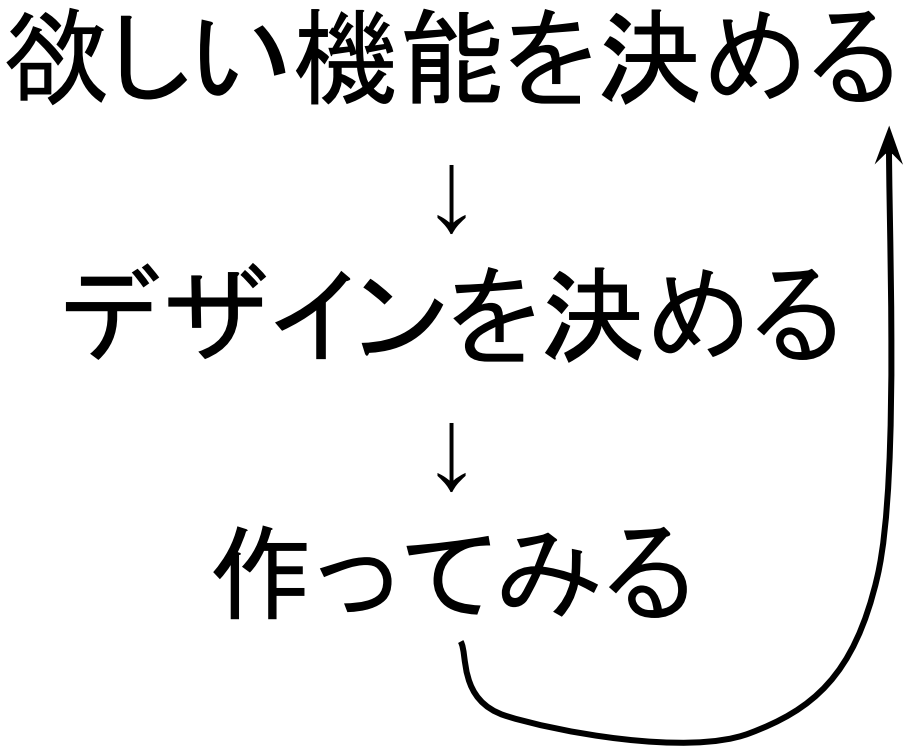
とはいえ

無理して使い辛いものを使い続けるのも無駄

最初に色々試すのが良いかも

慣れてくればコストの見積りが出来るようになる。慣れるほどの時間は無い。

まとめ


欲しい機能を決める
↓
デザインを決める
↓
作ってみる

目黒研究室 基礎技術講座 第二回

デザインからプロトタイピングまで

blanktar.jp

Thank you for listening!

Slack登録してない方
もし居たらお願いします

goo.gl/3YRc8d

本日の資料はこっち

goo.gl/h8Kp7T

Slackはこっち

goo.gl/3YRc8d

Thank you for listening!