



デザイナーの
提案する
Android
フロントエンド
実装

Lady Beetle Design

秋葉ちひろ

自己紹介

- 秋葉(とみた)ちひろ
- Lady Beetle Design



- Webデザイン
- HTML / CSS
- JavaScriptを少々
- アプリデザイン
- Android歴は1年半ぐらい、新入りです



自己紹介



CSS3デザイン
プロフェッショナルガイド



Web Designing

きっかけ



すぎもてさん

[@yoshi_rr](#)

Androidアプリは
とにかくダサい！

ダサすぎてほんとに
ダメなんですよ！

きっかけ



シャープハッカソン



Google主催 EXP-ハッカソン

- デザイナーとして参加
- 自分のわからないところで進んでいない
- エンジニアさんと連携がとれない



プログラムはできないけど
少しでも近づきたい！



赤井さん
@akai t

XML

おしえてー！



ITproでの連載

<http://itpro.nikkeibp.co.jp/article/COLUMN/20120113/378483/>

- デザイナーからの視点
- 公開当日、アクセスランキング第2位を獲得



001
WAKAYAMA

DESIGNERS HACK!

デザイナーが集まって
使いやすいアプリを設計・開発しよう!



おしゃれな空間で



自然に集まれて



楽しく開発



デザイナーズ
ハックとは?

デザイナー、エンジニア、UI/UXデベロッパーがメインとなり、アプリのUI/UX・デザインを徹底的に追及します。最終的にシステム実装前までのモックを作り上げることをゴールとします。
※必要に応じてシステムを組み込むこともあるでしょう。

普段の案件中でおおそかにされがちなUI/UXの企画設計。良いデザインを作っても、仕様が決まらないために何度も差し戻し。。そしてなんだか使いにくいアプリに...
そうならないために、UI/UXとデザインがいかに大切かを考えていく。それとともに、デザイナーとしての地位を確立していきたい。



主な流れ

① チームに分かれて題材を決める

分野が固まらないようにチームに分かれて、何について考えるかを決めます。



デザイナー



UI/UX 設計



エンジニア

- デザイナーが集まって
使いやすいアプリを
設計・開発する

デザイナーズハック

<http://designers-hack.net/>

現状

- 残念ながら
「このデザインなら使いたい！」
と思えるようなものがあまりない
- デザイナー入ってないんじゃないの？
- 少なくとも連携とれてないよね



もくじ

- 提案できるようになる
- できることから
はじめる
- Androidフロントエン
ド実装入門
- 気持ちの良いUIの提案



提案できる ようになる

もう、

言いたい放題にはさせないぞ！

- とってくるのはエンジニアさん。
話をしにいくのもエンジニアさん。
- 機能面についての仕様はがっちり固めてくるが、大事なところが抜けている

デザインを 作る上で 必要なもの

- デザインテイスト
- 色味
- ロゴ
- アプリのコンセプト

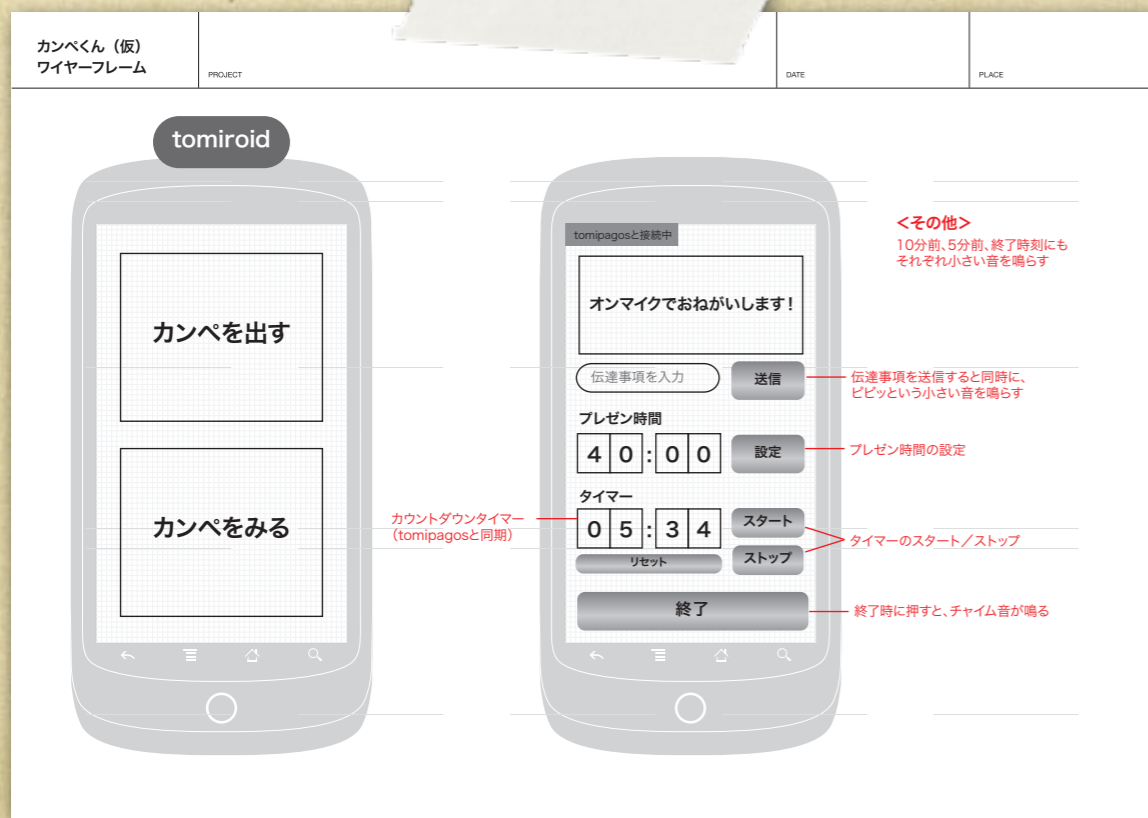


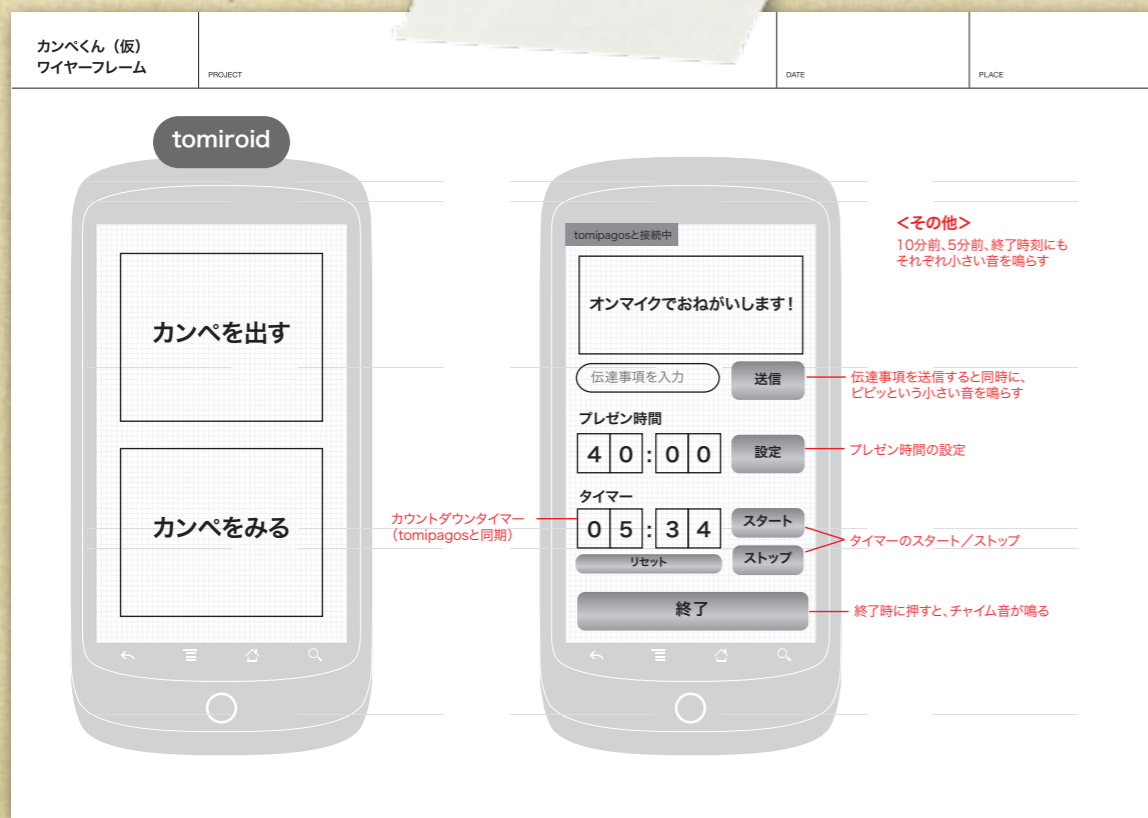
使い勝手

- ワイヤードesign



• ワイヤーデザイン





• ワイヤーデザイン

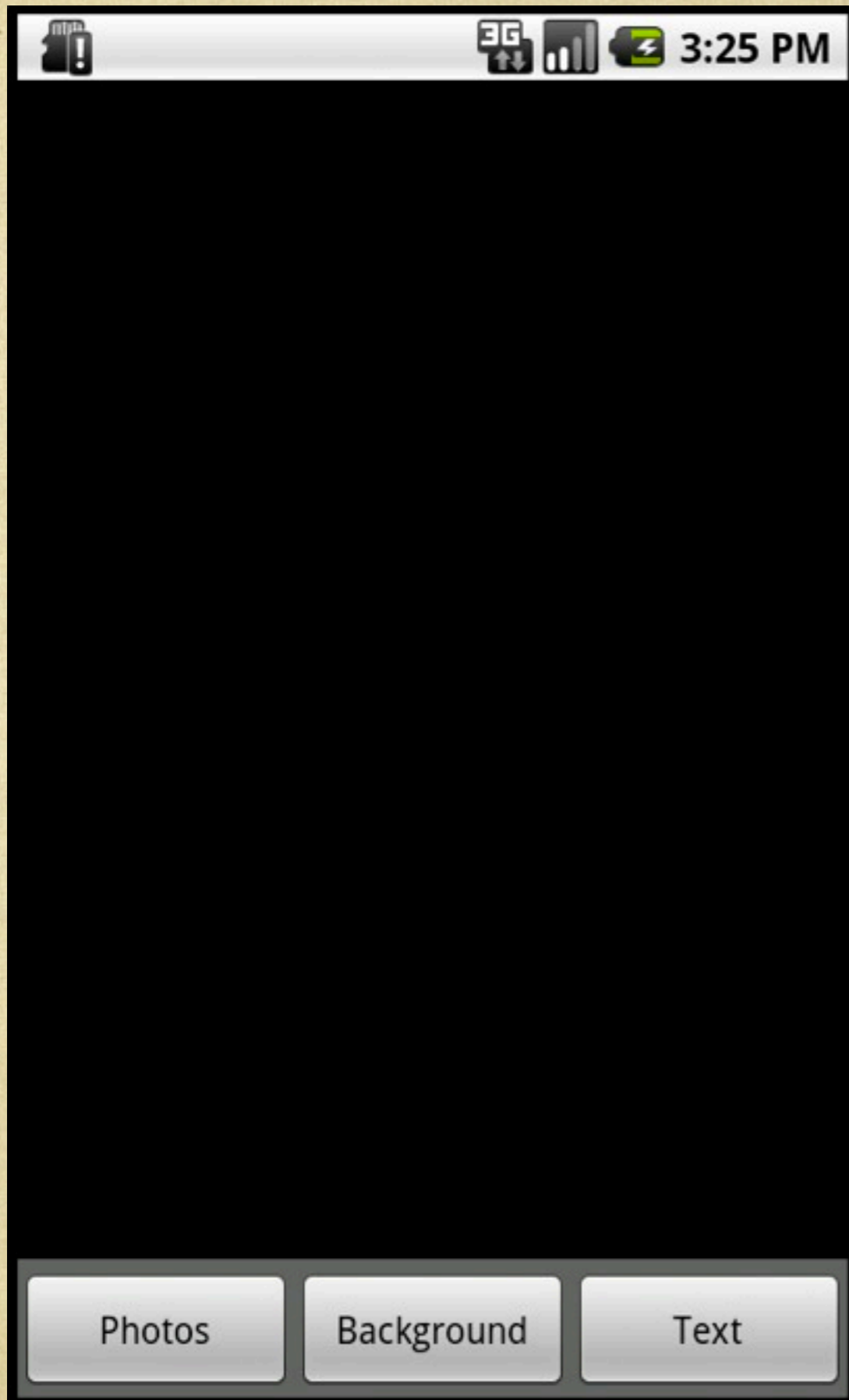
ユーザー インターフェース

ユーザーインターフェース(UI)

- 機械と利用者の間での情報をやりとりするための表示様式・方法
- ユーザのデータ入力方式を規定する、コンピュータシステムの「操作感」
- UI専門の人にまかせてもいいぐらい、重要な要素

よくある 使いにくい アプリ

- 操作方法がわからない
- 迷う
- 自分のいる場所がわからない



- 操作方法がわからない
- 迷う
- 自分のいる場所がわからない

提案できる ように なろう！

- 言われたものを作るだけでは向上しない
- デザインの視点、UIの視点は絶対に必要
- 今までのアプリを覆すすばらしいものを！



できることから はじめる

少しずつでいいんです



自分に できること

- デザイン
- HTML / CSS
- JavaScriptを少々



Android SDKを 使わず開発、 あとでpkg

- PhoneGap
- Titanium Mobile
- Flash
- Unity
- 他いろいろ…



Android SDKで Webの知識で やりきる

- WebView



Android SDKで レイアウトぐらい いけるやろ

できることから始める

- HTMLとXMLは似ている
- レイアウトまでならいけるんちゃう？



Webサイトの場合



Visual Design



Coding

HTML

CSS



System



Androidアプリの場合



Visual Design



Coding

XML



System



Webサイトの場合



Visual Design



Coding

HTML

CSS



System



Androidアプリの場合



Visual Design



Coding

XML



System

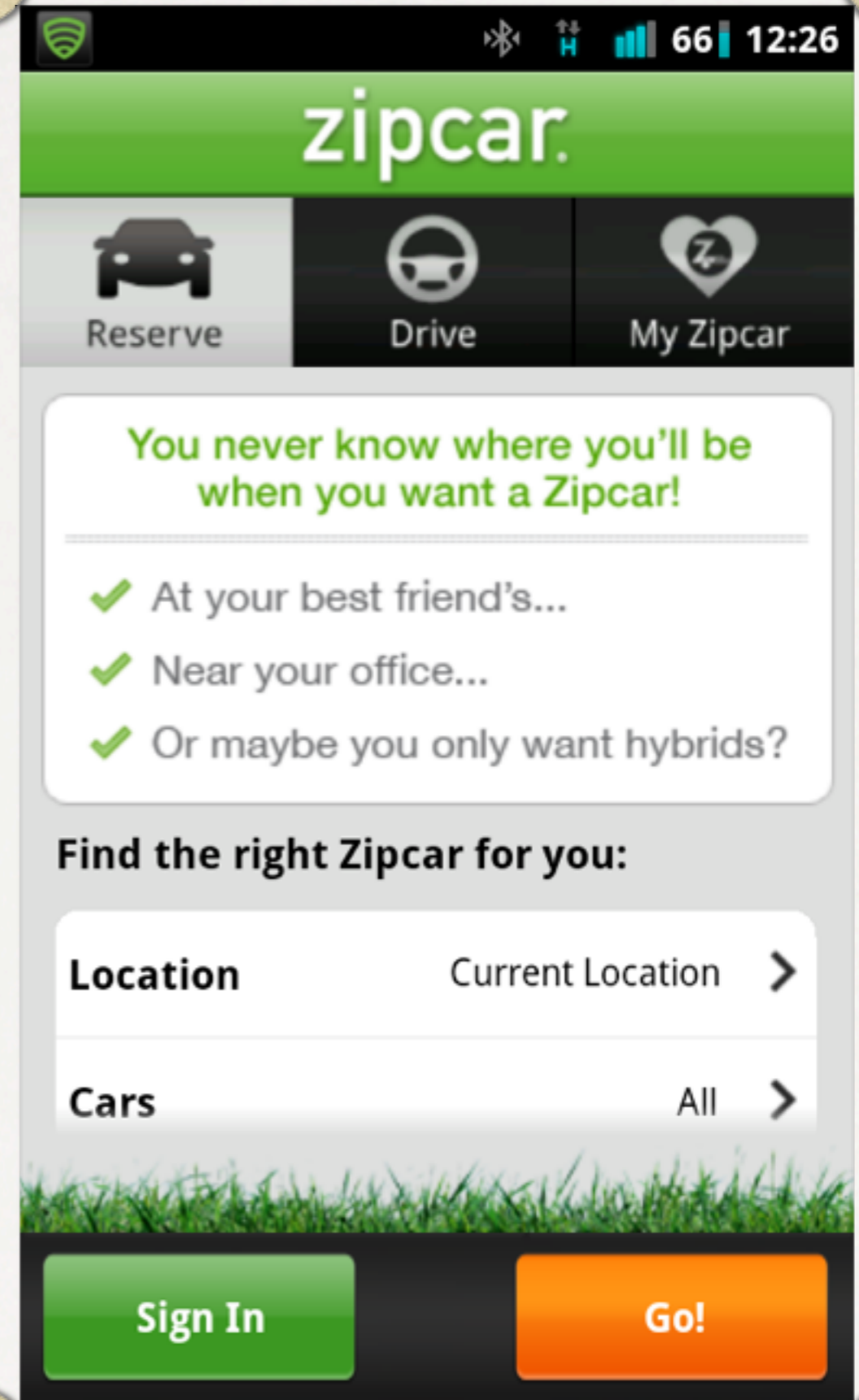
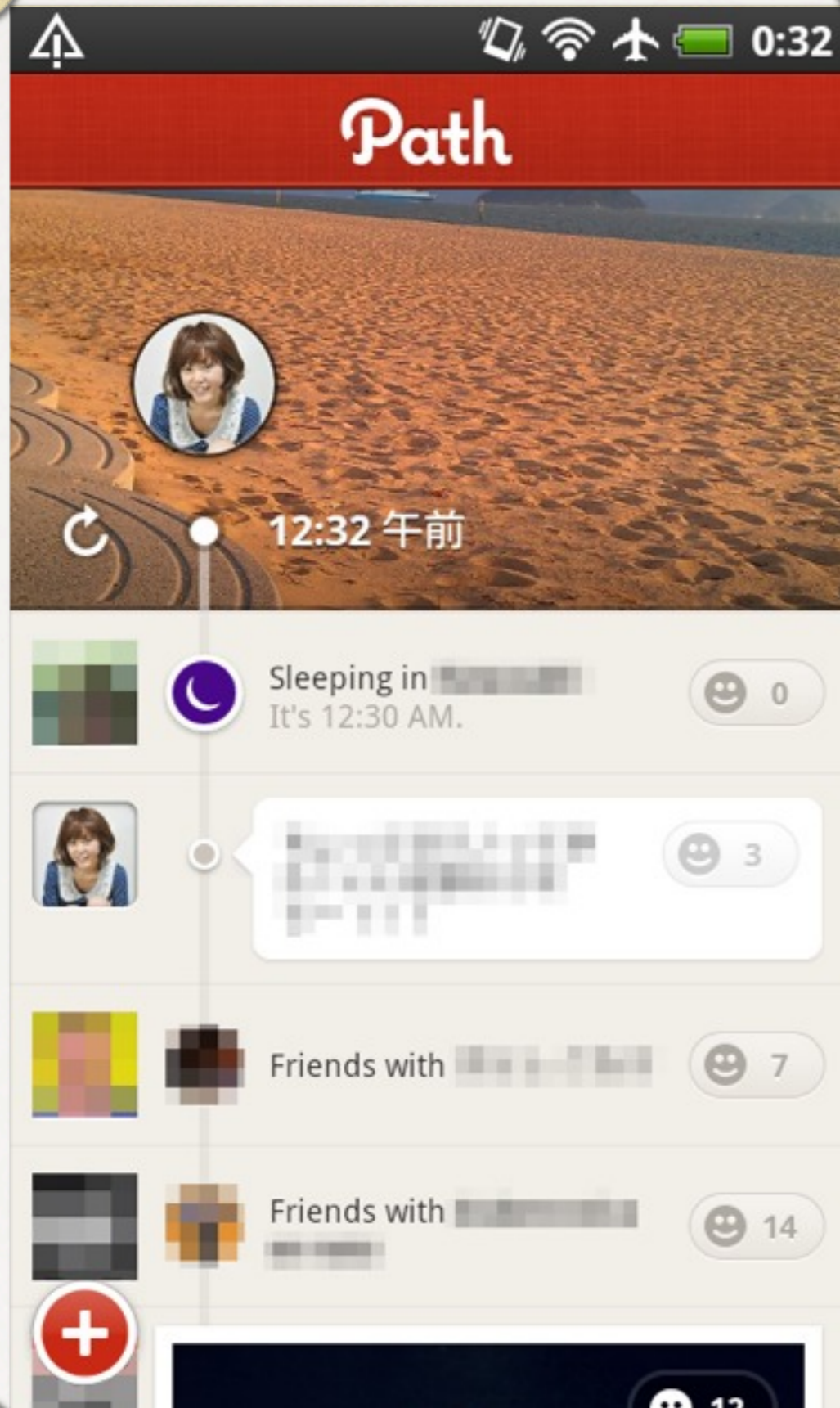
デザイナー の嘆き

解像度の関係で、
余白がバラバラに
なるねんて。

デザインしたあと、
実装するときに勝手に
作りかえられてん。

わりと軽視 されがちな デザイン

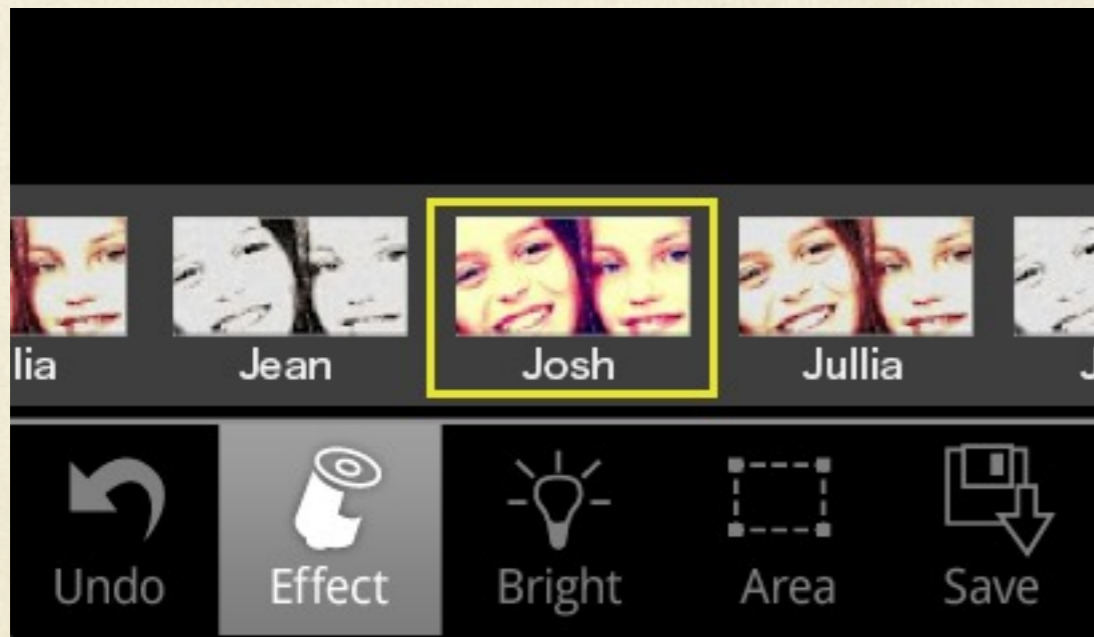
- 1pxのライン
- 余白の幅の統一
- 影の描画
- グラデーション
- テクスチャの使用
- 文字のジャンプ率





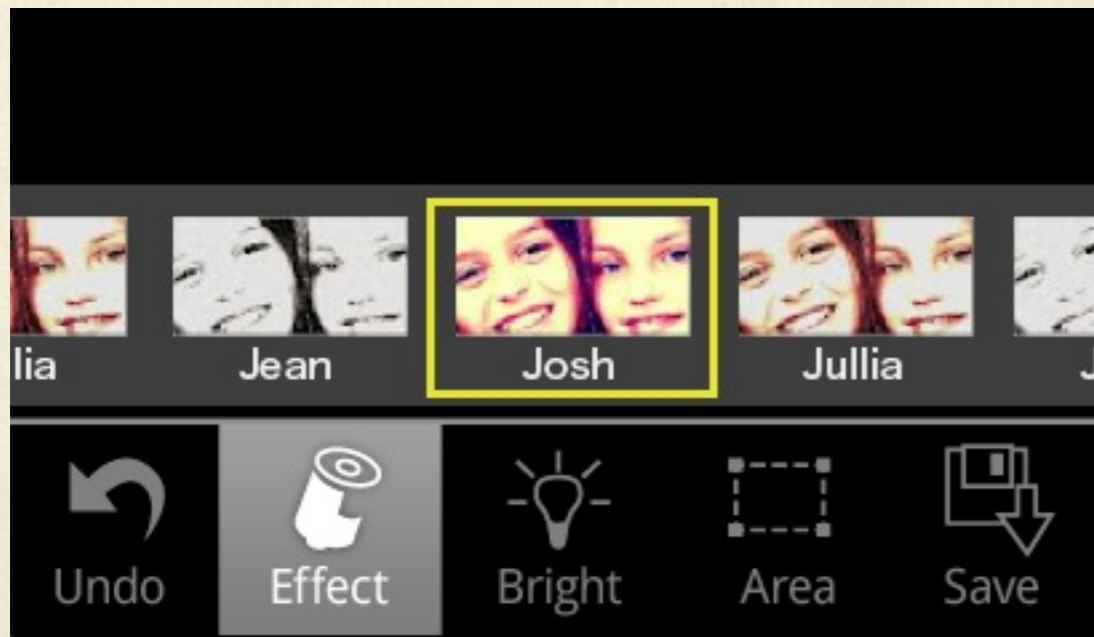


葬式UI





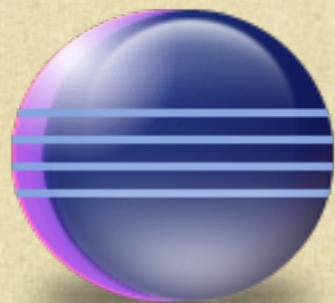
葬式UI





Android フロントエンド 実装入門

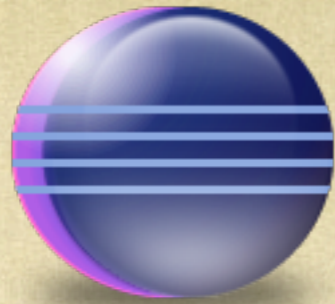
デザイナーが提案するよ！



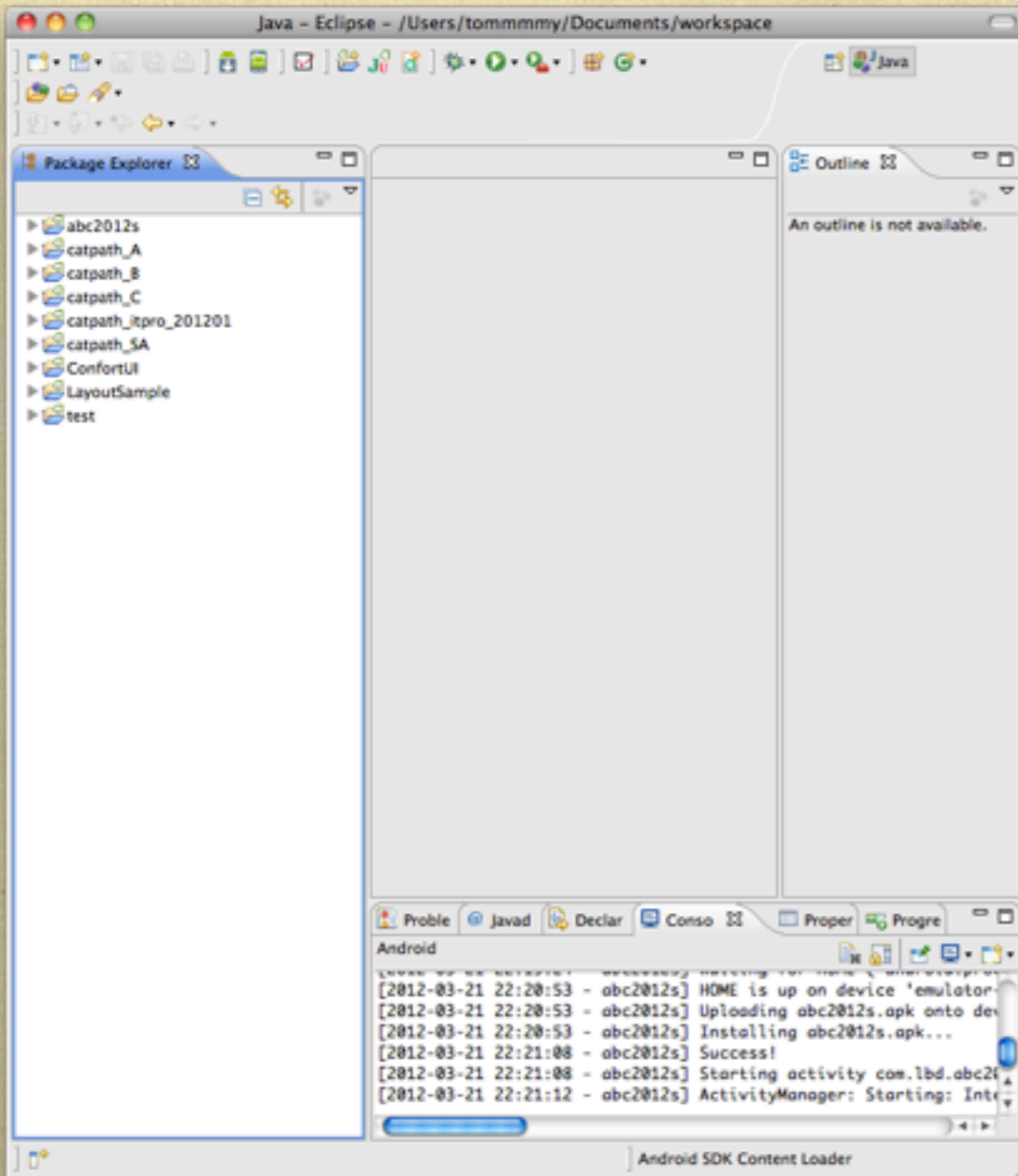
Eclipse



Android SDK



Eclipse



Android SDK





Package Explorer

- catpath_A
- catpath_B
- catpath_C
- catpath_itpro_201201
- catpath_SA
- ConfortUI
- LayoutSample
- test

New Android Project

Creates a new Android Project resource.

Project name:

Contents

- Create new project in workspace
- Create project from existing source
- Use default location

Location:

- Create project from existing sample

Samples:

Build Target

Target Name	Vendor	Platform	API Le
<input checked="" type="checkbox"/> Android 2.1-update1	Android Open Source Project	2.1-update1	7
<input type="checkbox"/> Android 2.2	Android Open Source Project	2.2	8
<input type="checkbox"/> Android 2.3.3	Android Open Source Project	2.3.3	10
<input type="checkbox"/> Android 3.0	Android Open Source Project	3.0	11
<input type="checkbox"/> Android 3.1	Android Open Source Project	3.1	12
<input type="checkbox"/> Android 3.2	Android Open Source Project	3.2	13

Standard Android platform 3.2

Properties

Application name:

Package name:

- Create Activity:

Min SDK Version:

Working sets

- Add project to working sets

Working sets:

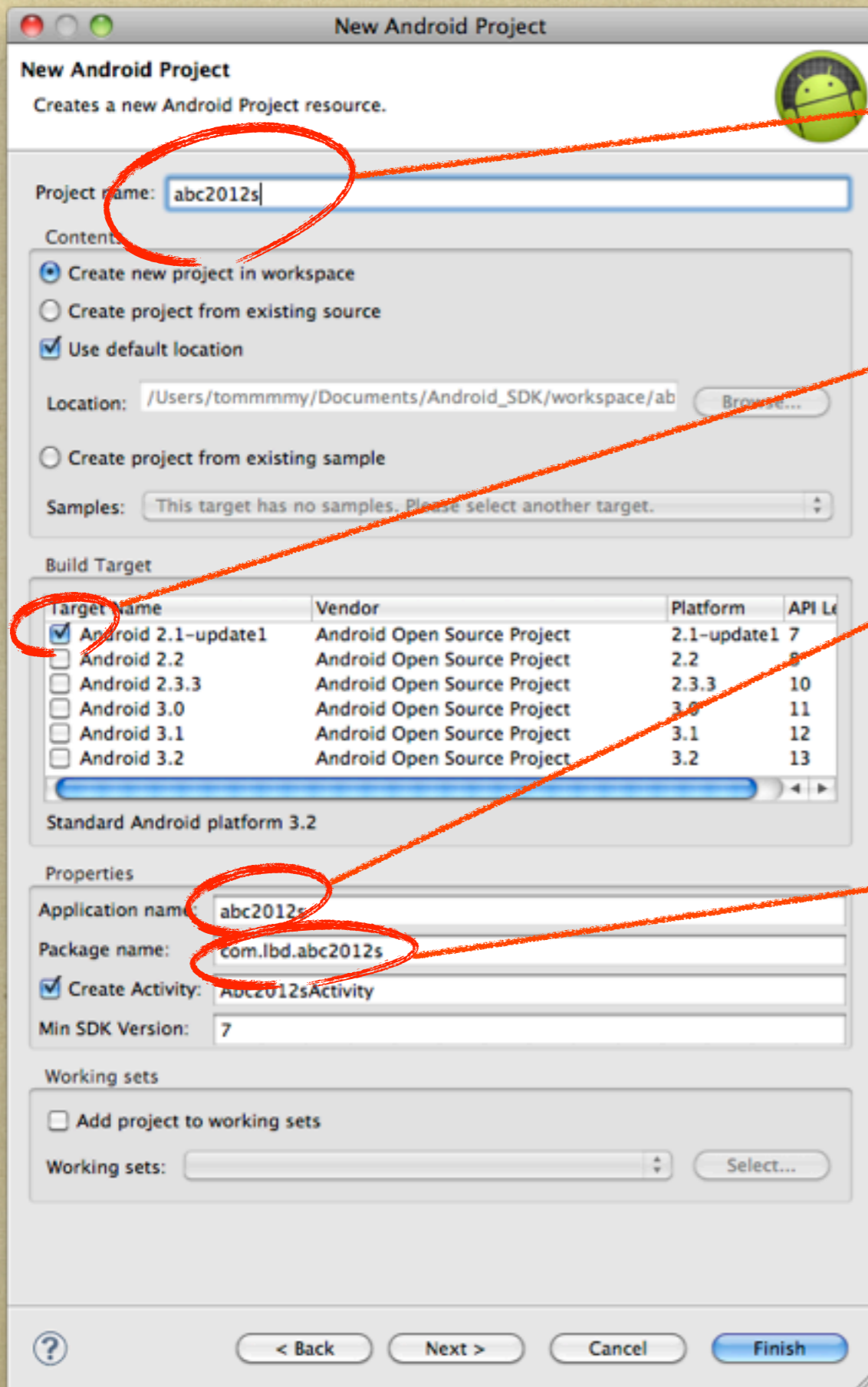
Java

Outline

An outline is not available.

Path	Location	Type

Android SDK Content Loader



Project name

Eclipse上のプロジェクトネーム

Build Target

Android 2.1-update1

Application name

アプリアイコンの下に出る
アプリ名

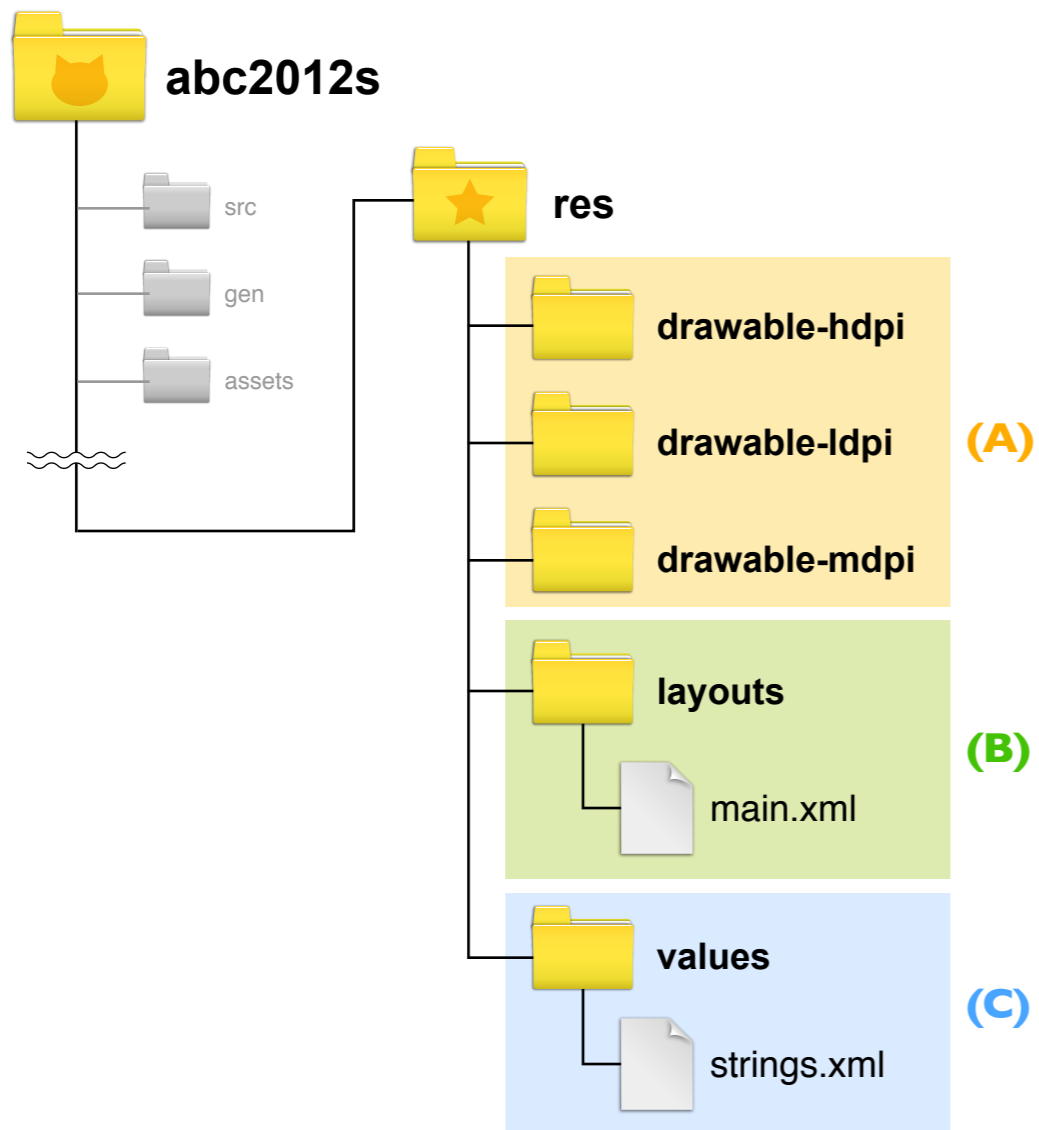
Package name

表には出ないが、Google Playの
URLに含まれることになる

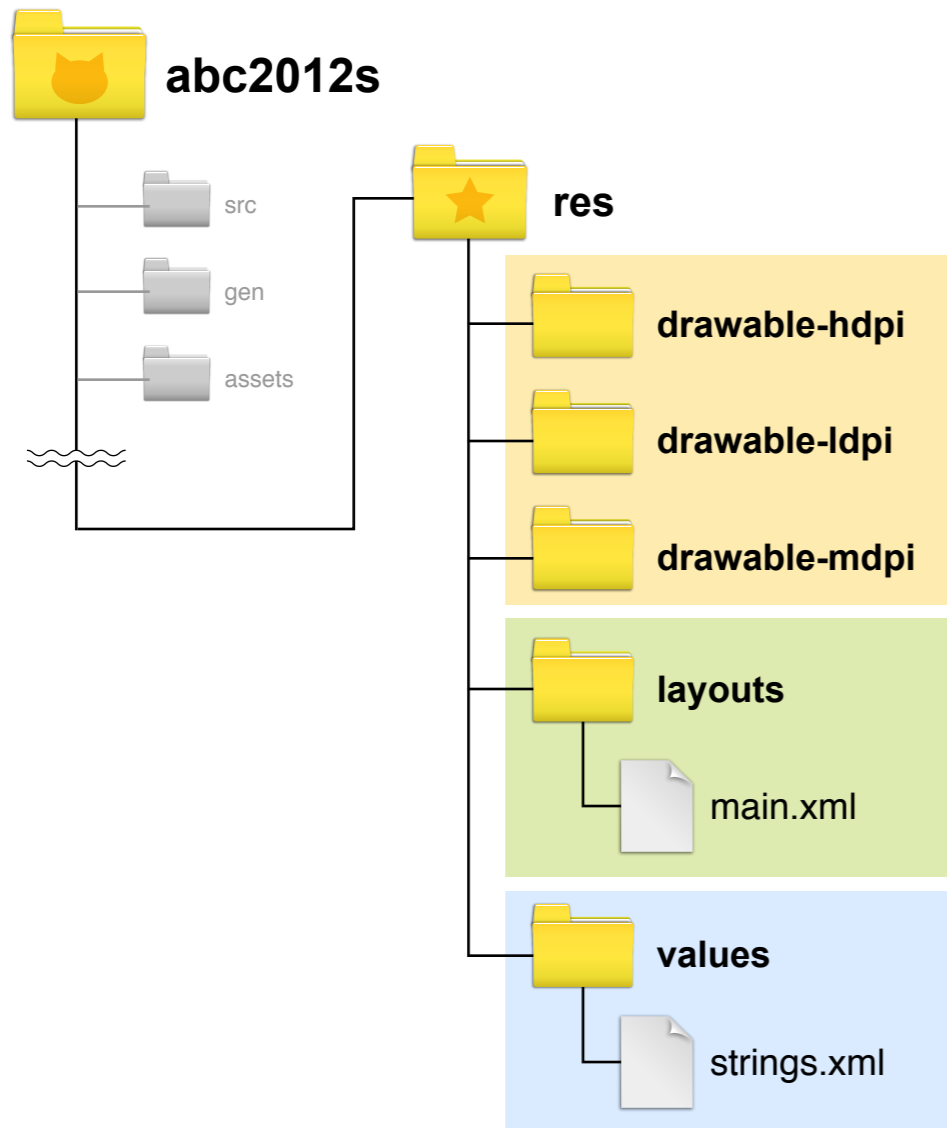
com . lbd . abc2012s

ドメイン名などを
逆にする(らしい)

アプリ名



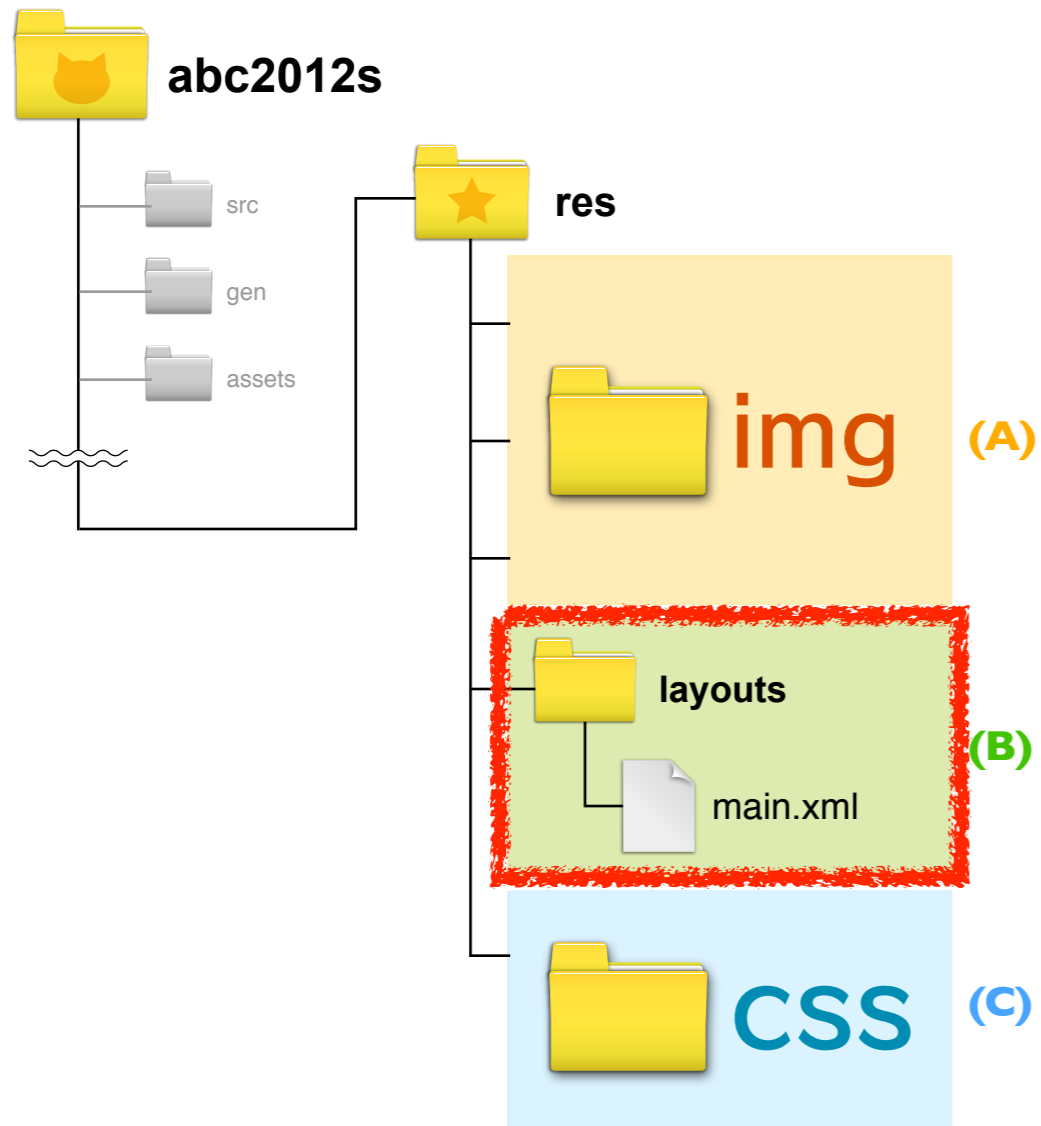
使うのは
res
||
resource



(A)画像格納場所

(B)レイアウトXML

(C)スタイルXML



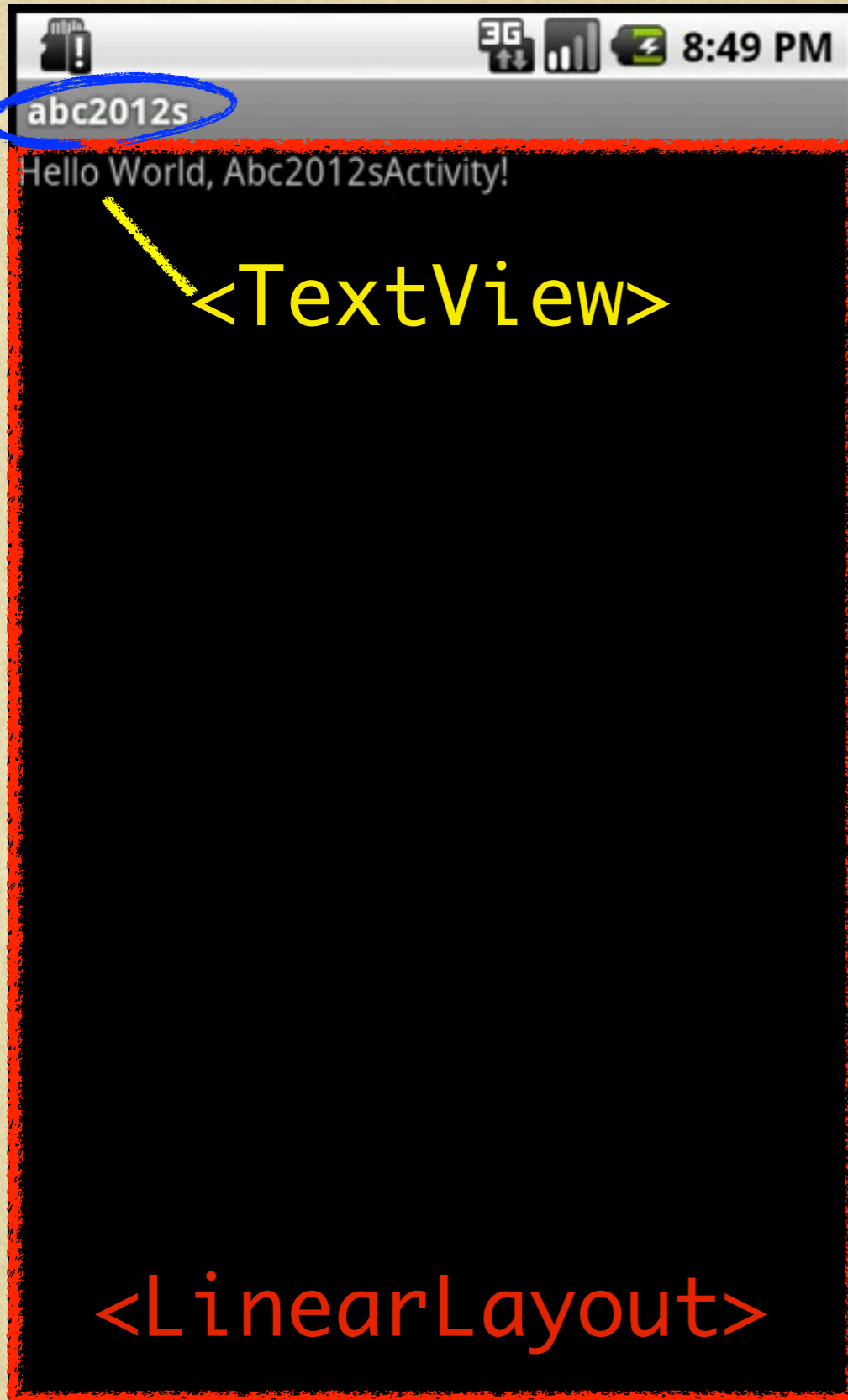
(A)画像格納場所

(B)レイアウトXML

(C)スタイルXML



main.xml



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="vertical"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  >
  <TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
  />
</LinearLayout>
```



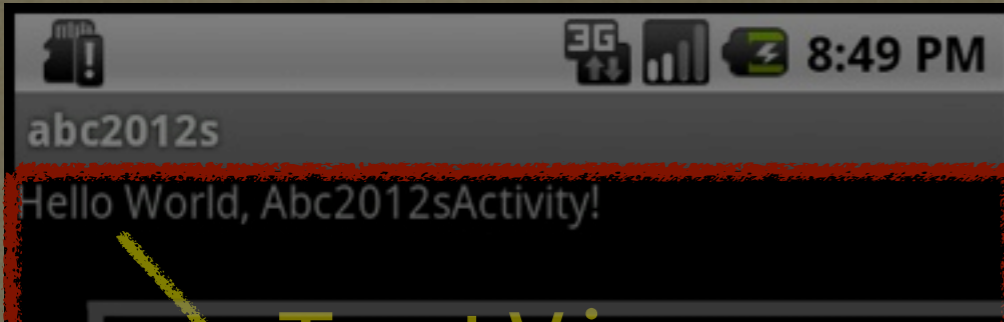
main.xml



```
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout  
    xmlns:android="http://schemas.android.  
        com/apk/res/android"  
    android:orientation="vertical"  
    android:layout_width="fill_parent"  
  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:text="@string/hello"  
    />  
</LinearLayout>
```



main.xml



```
android:text="@string/hello"
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"
```

```
<TextView
```

```
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"
```

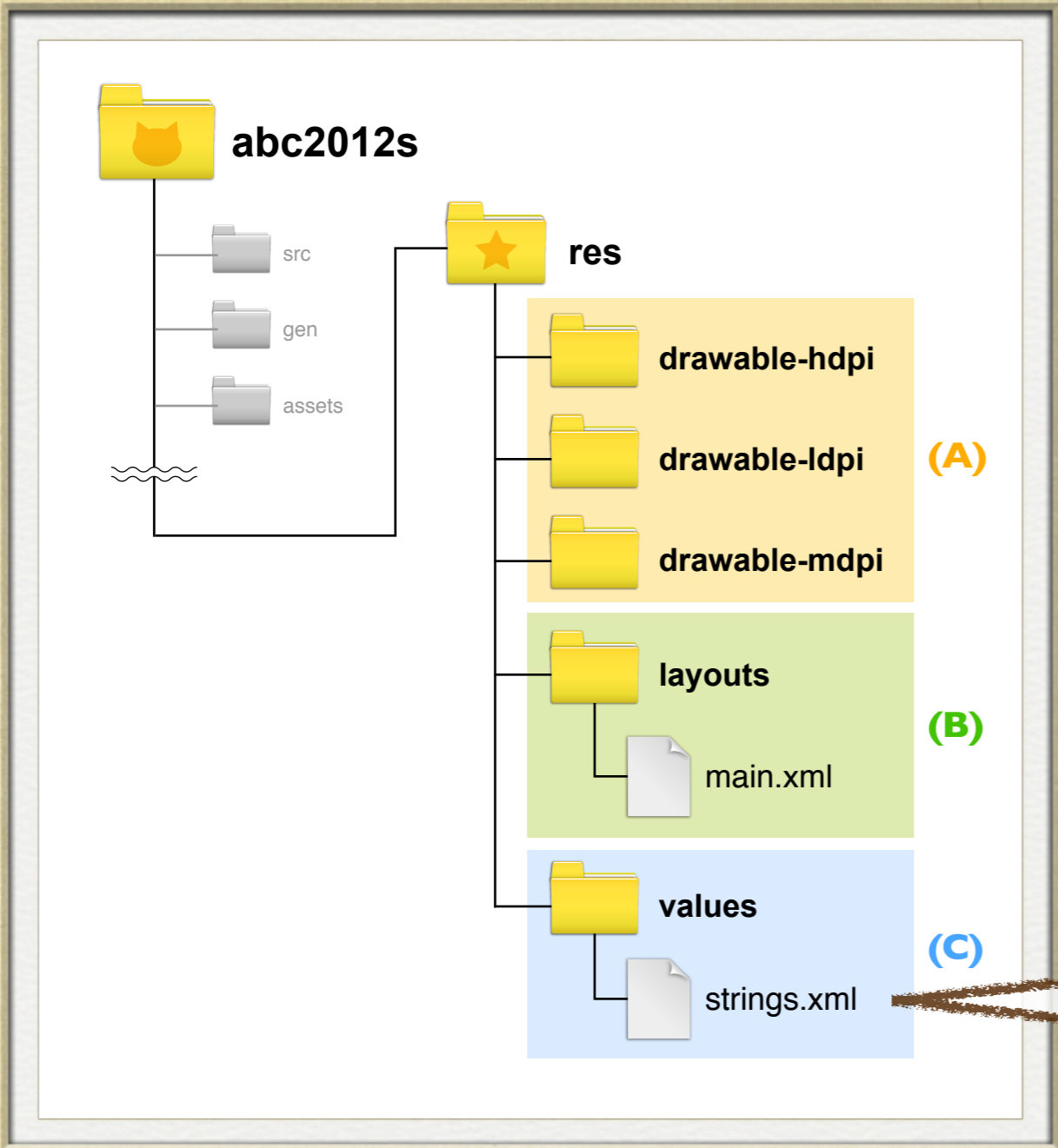
```
    android:text="@string/hello"
```

```
>
```

```
</LinearLayout>
```

```
<LinearLayout>
```

stringというファイルの
「hello」という場所に
内容を書いてあるよ！



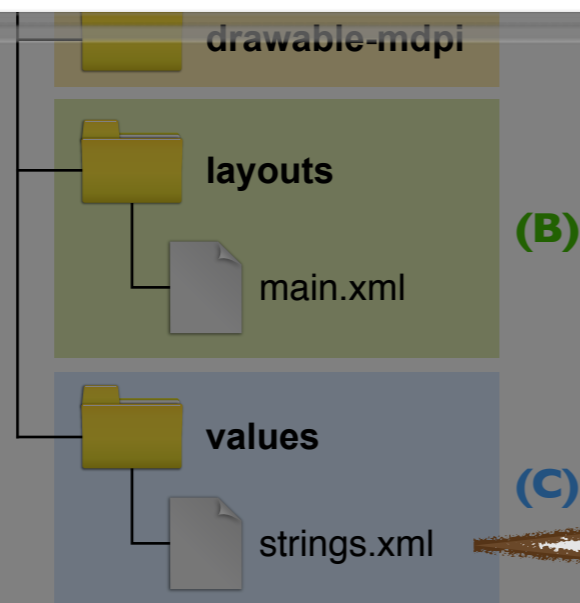
```

<string name="hello">
Hello World,
Abc2012sActivity!
</string>

```



```
android:text="@string/hello"
```



```
<string name="hello">  
Hello World,  
Abc2012sActivity!  
</string>
```

決まり 〈単位〉

~~px~~

dp

横幅や縦幅

sp

文字の大きさ

dp

横幅や縦幅

Density-independent Pixels

sp

文字の大きさ

Scale-independent Pixels

解像度に依存しないピクセル



**Galaxy S II
(docomo)**

480 × 800
(px)

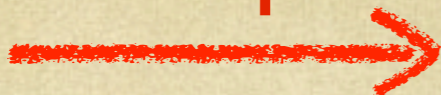


**HT-03A
(docomo)**

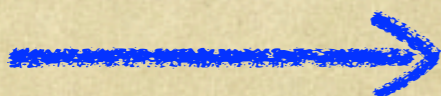
320 × 480
(px)



480px

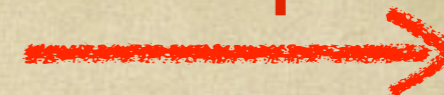


Galaxy S II
(docomo)

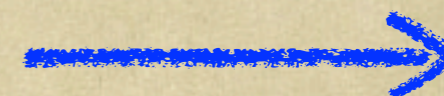


320dp

320px



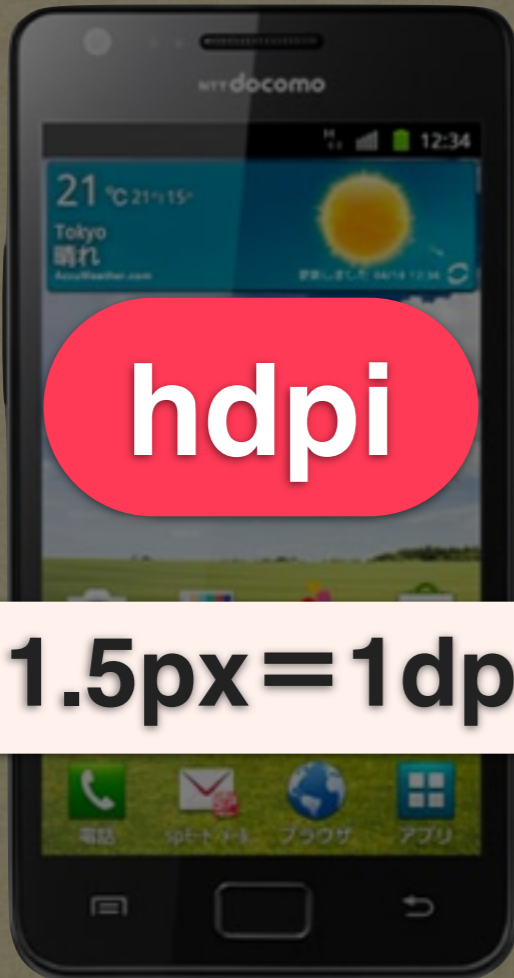
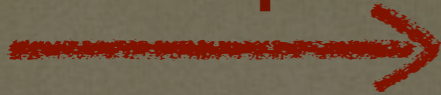
HT-03A
(docomo)



320dp



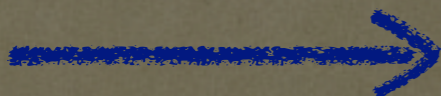
480px



hdpi

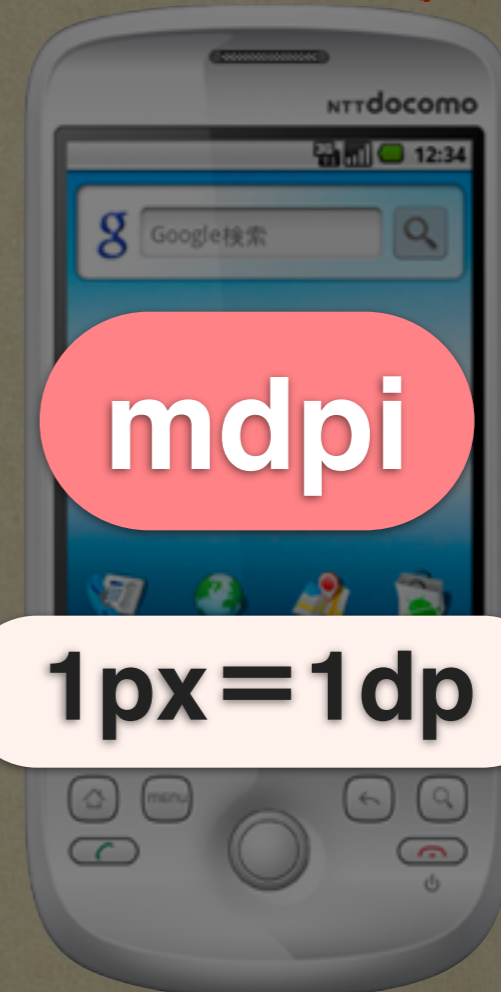
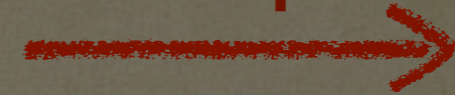
1.5px = 1dp

Galaxy S II
(docomo)



320dp

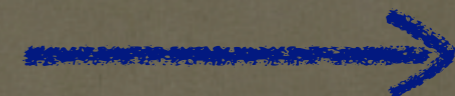
320px



mdpi

1px = 1dp

HT-03A
(docomo)



320dp

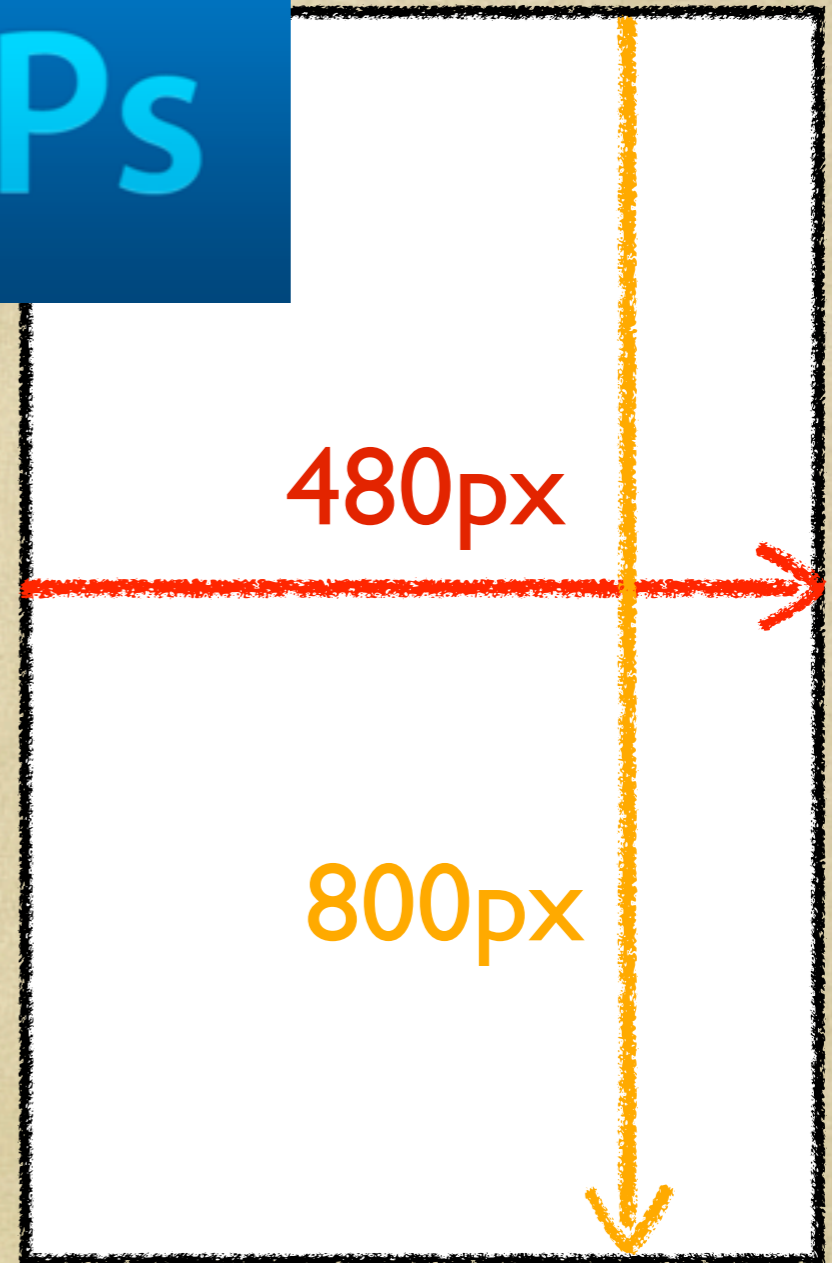


Galaxy S II
(docomo)

hdpi

mdpi

ldpi



決まり

<ファイル>

- 共通する記述はできるだけ省く
- 他言語化対応しやすくする



res



drawable-hdpi



drawable-mdpi



drawable-ldpi



layouts



main.xml



values



strings.xml



res



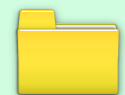
drawable-hdpi



drawable-mdpi



drawable-ldpi



layouts



main.xml



values



strings.xml



dimens.xml



colors.xml



styles.xml

言語ごとのテキストファイル

フォントサイズだけをまとめたファイル

色の指定だけをまとめたファイル

その他の共通するスタイルをまとめたファイル



res



drawable



drawable-hdpi



drawable-mdpi



drawable-ldpi



layouts



main.xml



values



strings.xml



dimens.xml



colors.xml



styles.xml

背景の指定だけをまとめた
ファイル

決まり 〈文法〉

- 書き方はXMLと似ている
- 属性が
`android:xxxx="-----"`



main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello"
    />
</LinearLayout>
```



main.xml

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
<?xml version="1.0"
      encoding="utf-8"?>
```

```
<TextView
```

```
    android:layout_width="fill_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="@string/hello"
```

```
    />
```

```
</LinearLayout>
```



main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  <LinearLayout
    android:layout_height="fill_parent"
  >
  <TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
  />
</LinearLayout>
```



main.xml

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
xmlns:android="http://  
schemas.android.com/apk/res/  
android"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="@string/hello"
```

```
  />
```

```
</LinearLayout>
```



main.xml

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
    android:orientation=  
        "vertical"
```

```
<TextView
```

```
    android:layout_width="fill_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="@string/hello"
```

```
</>
```

```
</LinearLayout>
```



main.xml

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
    android:layout_width=  
        "fill_parent"
```

```
<TextView
```

```
    android:layout_width="fill_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="@string/hello"
```

```
</>
```

```
</LinearLayout>
```



main.xml

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
    android:layout_height=  
        "fill_parent"
```

```
<TextView
```

```
    android:layout_width="fill_parent"
```

```
    android:layout_height="wrap_content"
```

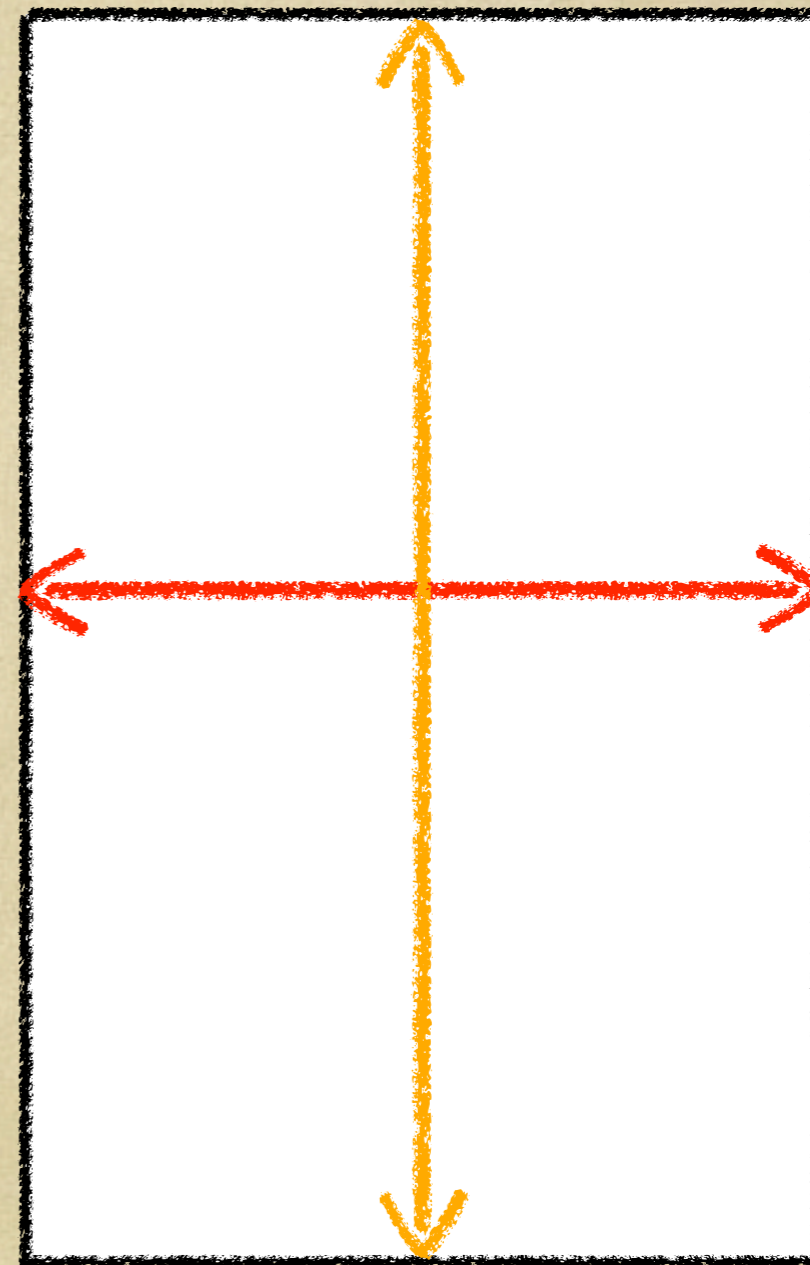
```
    android:text="@string/hello"
```

```
</>
```

```
</LinearLayout>
```


"fill_parent"

いっぱいいっぱいに広げる



"wrap_content"

その部分だけにする





main.xml

撮影する

ギャラリーから選ぶ

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="vertical"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  >
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="撮影する"
    />
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="ギャラリーから選ぶ"
    />
</LinearLayout>
```



main.xml

撮影する

ギャラリーから選ぶ

```
<Button
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="撮影する"
```

```
</>
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.
        com/android"

```

```
    android:layout_height="wrap_content"
```

```
    android:text="ギャラリーから選ぶ"
```

```
</>
```

```
</LinearLayout>
```



main.xml

撮影する

ギャラリーから選ぶ

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="vertical"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  >
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="撮影する"
    />
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="ギャラリーから選ぶ"
    />
</LinearLayout>
```

main.xml

撮影する

ギャラリーから選ぶ

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="vertical"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  >
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="撮影する"
    />
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="ギャラリーから選ぶ"
    />
</LinearLayout>
```

main.xml

撮影する

ギャラリーから選ぶ

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="horizontal"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  >
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="撮影する"
    />
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="ギャラリーから選ぶ"
    />
</LinearLayout>
```



main.xml

撮影する

ギャラリーから選ぶ

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="horizontal"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  >
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="撮影する"
    />
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="ギャラリーから選ぶ"
    />
</LinearLayout>
```




main.xml

撮影する

ギャラリーから選ぶ

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="horizontal"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  >
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="撮影する"
    />
  <Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="ギャラリーから選ぶ"
    />
</LinearLayout>
```



main.xml

撮影する

ギャラリーから選ぶ

```
android:layout_weight="1"
android:layout_width="0dp"
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:orientation="horizontal"
  android:layout_width="fill_parent"
    fill_parent"
    "wrap_content"
    ="wrap_content"
  android:text="撮影する"
  />
<Button
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:text="ギャラリーから選ぶ"
  />
</LinearLayout>
```

決まり

< 9-patch >

- 伸びる可能性のあるものは、9スライスを設定したものを使う
- 角丸の背景

撮影する



ギャラリーから選ぶ

- 文字量によって伸び縮みする
- 端末によって大きさが変わる



撮影する

9-patch



グラデーションのボタン



横に伸びるのは問題なし



縦に伸びると変なグラデーションに…


```
<gradient
```

```
    android:startColor="#f28d41"
```

```
    android:endColor="#713000"
```

```
    android:angle="270"
```

```
>
```



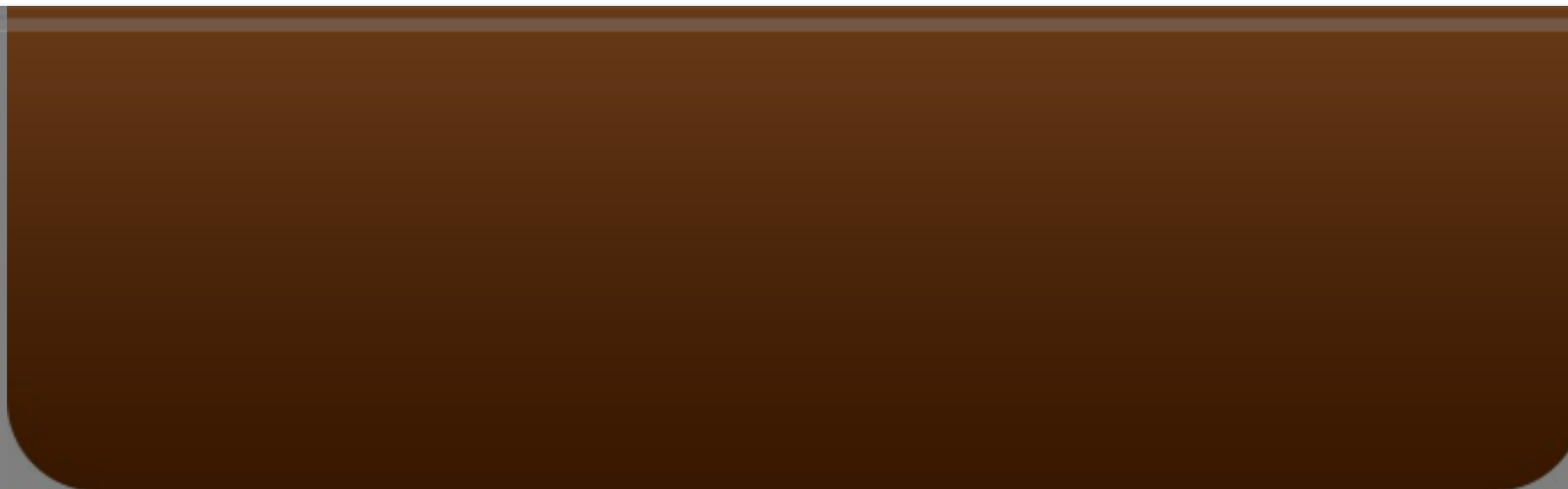
```
<stroke
```

```
    android:width="0.66dp"
```

```
    android:color="#af6027" />
```



```
<corners  
    android:radius="10dp" />
```





res



drawable



drawable-hdpi



drawable-mdpi



drawable-ldpi



layouts



main.xml



values



strings.xml



dimens.xml



colors.xml



styles.xml

背景の指定だけをまとめた
ファイル



bg_gradbtn.xml

```
<gradient />  
<stroke />  
<corners />
```



res

```
<TextView
```

```
    android:text="撮影する"
```

```
    android:background=
```

```
        "@drawable/bg_gradbtn"
```

```
>
```

 **bg_gradbtn.xml** strings.xml dimens.xml colors.xml styles.xml

```
android:background="@drawable/bg_title"
```

```
android:background="#fcf9f0"
```

```
android:paddingLeft="10dp"
```

```
android:layout_marginLeft="10dp"
```

```
android:shadowDx="0.66"
```

```
android:shadowDy="0.66"
```

```
android:shadowRadius="0.66"
```

```
android:shadowColor="#6000"
```

```
<TextView  
    style="@style/kind"  
    android:text="キジ"  
    android:id="@+id/kind1"  
    android:layout_toRightOf="@id/friendPhoto"  
    android:l
```

```
/>
```

```
<TextView
```

```
    style="@
```

```
    android:
```

```
    android:
```

```
    android:
```

```
/>
```

```
</RelativeLayout
```

```
nearLayout>
```

- Ⓐ android:lines
- Ⓐ android:linksClickable
- Ⓐ android:lineSpacingExtra
- Ⓐ android:lineSpacingMultiplier
- Ⓐ android:longClickable
- Ⓐ android:layout_toLeftOf
- Ⓐ android:layout_toRightOf
- Ⓐ android:layout_above
- Ⓐ android:layout_below
- Ⓐ android:layout_alignBaseline
- Ⓐ android:layout_alignLeft

Make
[inte



気持ちの良い UIの提案

使って楽しいものを！



今、Androidアプリに
欠けていること

気持ちの良い 動き

Grid Lens

Available on the App Store
\$0.99
FOR IPHONE

LET US INTRODUCE YOU
A **BRAND-NEW**
CAMERA!

1. GRID LENS LETS YOU HAVE YOUR PRECIOUS MOMENTS FROM DIFFERENT ANGLES ON A SINGLE FRAME.

2. YOU CAN TAKE A PHOTOS OF A SAME MOMENT FROM DIFFERENT ANGLE ON ONE FRAME.

TAP TAP TAP!

CAMERA MODE

- 1 SINGLE LENS MODE (DIVIDING UNITS)
- 2 MULTI LENS MODE (COMPOUND EYES-REPEATING UNITS)
- 3 BURST SHOT
 - 1 MANUAL SHOT
 - 2 TOUCHING TO CAPTURE ON GRID
 - 3 MODIFYING THICKNESS OF GRID LINE
 - 4 PICKING GRID BORDER COLORS
 - 5 FRAME RATIO

TAP TO DIVIDE FRAME.



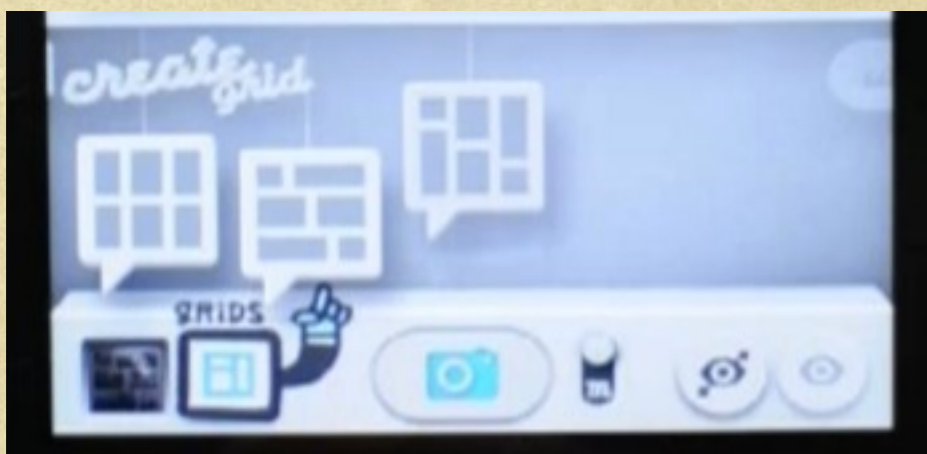
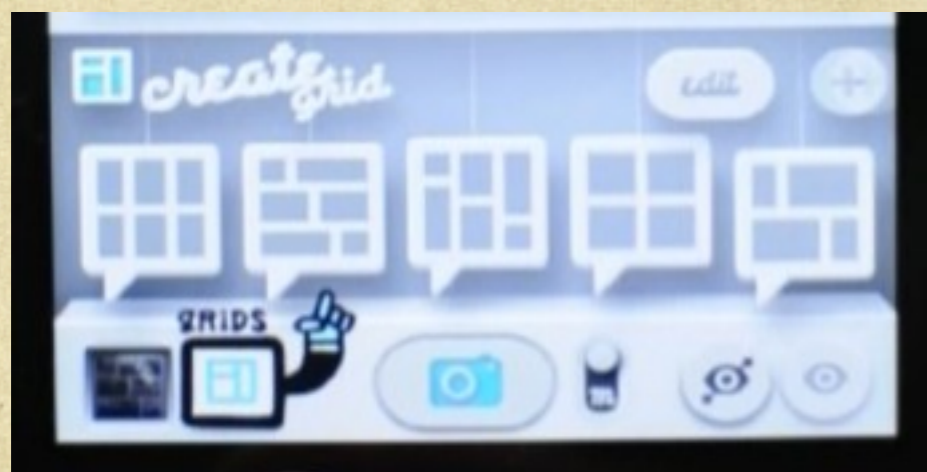
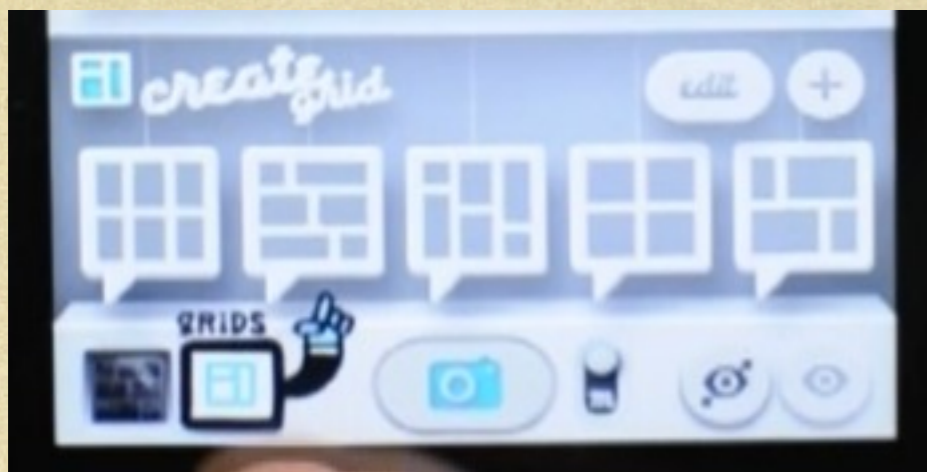
Grid Lens
¥85



気持ちの良いUIの提案

気持ちの良い 動き

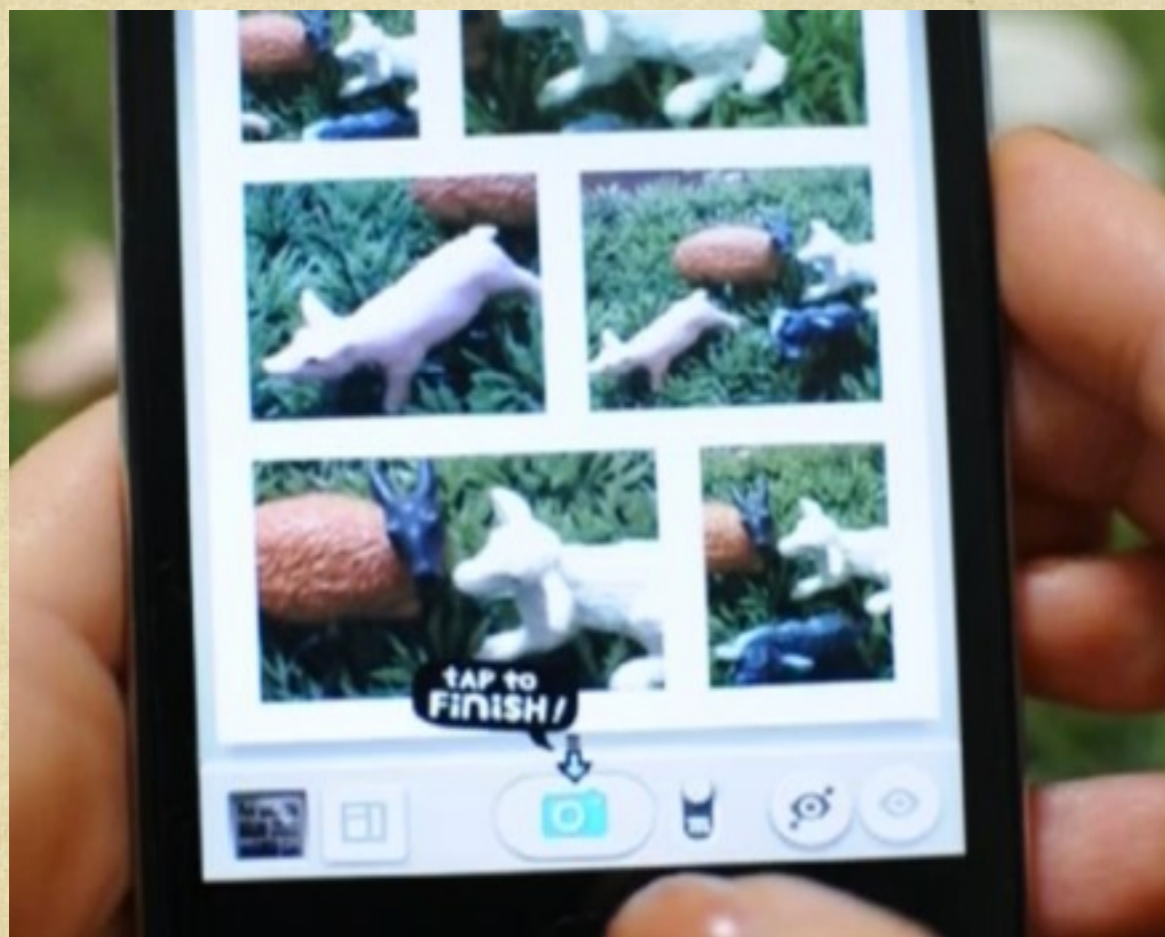
- ユーザーの邪魔にならないぐらいの単純なもの
- 重力を取り入れたような自然な動き (イーゼリング)



- ユーザーの邪魔にならないぐらいの単純なもの
- 重力を取り入れたような自然な動き (イーザリング)

ユーザーの 操作性

- 説明書がなくても使える
- 次にやるべきことがそこで明示される



- 説明書がなくても使える
- 次にやるべきことがそこで明示される

どうしても
やりたくて
お願いした



第4回シャープハッカソン@東広島

スタ☆三



@adakoda



@mgmix5



@cyberspacefarm



@tommmmy



@tunakko_



@muo_jp



@cattaka_net





@tommmmy

Grid Lens みたいな
動きにしたいんです！

いっすよ～
わかりました～



@cattaka_net





アニメーション

- XMLでかけます



res



drawable



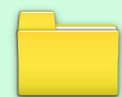
drawable-hdpi



drawable-mdpi



drawable-ldpi



layouts



main.xml



values



strings.xml



res



anim



drawable



drawable-hdpi




drawable-mdpi



drawable-ldpi




layouts



main.xml



values



strings.xml

アニメーションの設定を
まとめる

```
<set
  xmlns:android="http://schemas.android.
    com/apk/res/android"
  android:interpolator=
    "@android:anim/bounce_interpolator"
  android:fillAfter="true"
>
```

アニメーションの種類

バウンス

加速・減速

ため・はみ出し

layouts

main.xml

values

strings.xml

<scale

android:fromXScale="1.0"

android:fromYScale="1.0"

android:toXScale="2.0"

android:toYScale="2.0"

android:pivotX="0%"

android:pivotY="0%"

android:duration="300"

/>

大きさ

原点

時間

<alpha

android:fromAlpha="1.0"

android:toAlpha="0.0"

android:duration="700"

android:startOffset="300"

/>

アルファ

開始時間

main.xml



values



strings.xml

**res****anim****drawable****drawable-h****drawable-r****drawable-ldpi****layouts**

main.xml

**values**

strings.xml

`<scale />``<alpha />``<rotate />``<translate />`

の設定を

脱葬式UI





@adakoda

スライドバーも
変えられるよ♡



よわく



つよく



土下座

アップロード



デコメとして保存

共有する



まとめ

わたしたち から提案 できること

- デザイン
- 動き
- それが表現可能かどうかは相談すればよい

わたしたちにはしか
提案できないこと

使って楽しいアプリを
たくさんつくろう！

Android Layout Cookbook

アプリの価値を高める開発テクニック

SDK 2.3 対応
(Gingerbread)

あんどろいどゆき
[著]

デザインと使いやすさを徹底検証

「ただ動く」から「使いたくなる」
アプリへステップアップ!

見た目や
レイアウトの洗練
.....
ユーザビリティ、
パフォーマンスの向上
.....
マルチデバイス
への対応

ダウンロードして使える
サンプルコード付き!

インプレスジャパン

おすすめ本



ありがとうございました！

Lady Beetle Design

秋葉ちひろ