

実践 ANDROIDアプリ開発

@JX通信社

酒本伸也

2015/10/7

復習

RETROFIT

+

GOOGLE WEB SEARCH API

GOOGLE WEB SEARCH API

- Deprecated
 - サンプルとして使いやすかったため
 - Qiita APIと比較して検証


```
- responseData: {  
  - results: [  
    - {
```

web search api

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "",  
content: "2015年10月16日（金）東京・千駄  
10種類以上のフレーバーをたのしめる<b>アイ  
unescapeUrl: "http://www.mylohas.net/20  
url: "http%3A%2F%2Fwww.mylohas.net%2F  
title: "日本初上陸。乳製品も大豆も使わない「  
titleNoFormatting: "日本初上陸。乳製品も大豆  
location: "",  
publisher: "取材（一次情報メディア）（登録）",  
publishedDate: "Sun, 04 Oct 2015 15:41:15  
signedRedirectUrl: "http://news.google.com/  
O-XuV_Uu0ZVVw0QA",  
language: "ja",  
- image: {  
  url: "http://www.mylohas.net/images/2  
tbUrl: "http://t3.gstatic.com/images?q:  
originalContextUrl: "http://www.myloha  
publisher: "取材（一次情報メディア）（登  
tbWidth: 80,  
tbHeight: 43  
}
```

```
},
```

```
- {
```

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "http://news.google.com/news/s
```

```
- {
```

qiita api

```
body: "<p>こんにちは。エウセルソフトの田淵です。</p>  
</i></a>Now On Tap って？</h3> <blockquote>  
</blockquote> <p>Android 6 Marshmallow の(個人的  
りから離れずにカードビューで表示してくれる。という  
ことは、全部ホームボタンが教えてくれる。「Google Ne  
ってみる日本語版はまだか" class="fragment"></span>  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
<i class="fa fa-link"></i></a>使ってみる(日本語版はま  
</p> <h5> <span id="初回実行" class="fragment"></span>  
Tap はアプリ起動中にホームボタンを長押しするだけで  
href="http://cdn-ak.f.st-hatena.com/images/fotolife/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
ているところで Now on Tap ! "Xamarin Test Cloud rev  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
Blog の検索と、Website へのリンクが表示されると共に  
いるようです。</p> <p><a href="http://cdn-ak.f.st-h  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
(株)ナビゲート という Web制作会社が出てきます！(ど  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
てきますw </p> <p><a href="http://cdn-ak.f.st-hate  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
string の情報を引っ張って解析しているはず)、アプリ側  
さそうですね。</p> <p>以上です。</p> ",  
comment_count: 0,  
created_at_as_seconds: 1444141459,  
created_at_in_words: "約1時間",  
created_at: "2015-10-06 23:24:19 +0900",
```



```
- responseData: {  
  - results: [  
    - {
```

web search api

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "",  
content: "2015年10月16日（金）東京・千駄  
10種類以上のフレーバーをたのしめる<b>アイ  
unescapeUrl: "http://www.mylohas.net/20  
url: "http%3A%2F%2Fwww.mylohas.net%2F  
title: "日本初上陸。乳製品も大豆も使わない「  
titleNoFormatting: "日本初上陸。乳製品も大豆  
location: "",  
publisher: "取材（一次情報メディア）（登録）",  
publishedDate: "Sun, 04 Oct 2015 15:41:15  
signedRedirectUrl: "http://news.google.com/  
O-XuV_Uu0ZVVw0QA",  
language: "ja",  
- image: {  
  url: "http://www.mylohas.net/images/2  
tbUrl: "http://t3.gstatic.com/images?q:  
originalContextUrl: "http://www.myloha  
publisher: "取材（一次情報メディア）（登  
tbWidth: 80,  
tbHeight: 43  
}
```

```
},  
- {
```

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "http://news.google.com/news/s
```

```
[  
  - {
```

qiita api

```
body: "<p>こんにちは。エウセルソフトの田淵です。</p>  
link"></i></a>Now On Tap って? </h3> <blockquote  
</blockquote> <p>Android 6 Marshmallow の(個人的  
りかたで、ドキュメントが読める。という  
ことは、全部ホームボタンが教えてくれる。「Google Ne  
ってみる日本語版はまだか" class="fragment"></span>  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
<i class="fa fa-link"></i></a>使ってみる(日本語版はま  
</p> <h5> <span id="初回実行" class="fragment"></span>  
Tap はアプリ起動中にホームボタンを長押しするだけです  
href="http://cdn-ak.f.st-hatena.com/images/fotolife/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
ているところで Now on Tap ! "Xamarin Test Cloud rev  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
Blog の検索と、Website へのリンクが表示されると共に  
いるようです。</p> <p><a href="http://cdn-ak.f.st-h  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
(株)ナビゲート という Web制作会社が出てきます！(ど  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
てきますw </p> <p><a href="http://cdn-ak.f.st-hate  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
string の情報を引っ張って解析しているはず)、アプリ側  
さそうですね。</p> <p>以上です。</p> ",  
comment_count: 0,  
created_at_as_seconds: 1444141459,  
created_at_in_words: "約1時間",  
created_at: "2015-10-06 23:24:19 +0900",
```

配列から始まる


```
- responseData: {  
  - results: [  
    - {
```

web search api

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "",  
content: "2015年10月16日（金）東京・千駄  
10種類以上のフレーバーをたのしめる<b>アイ  
unescapeUrl: "http://www.mylohas.net/20  
url: "http%3A%2F%2Fwww.mylohas.net%2F
```

```
public interface CatNewsService {  
  
    @GET("/ajax/services/search/news?v=1.0")  
    void results(@Query("q") String keyword,  
                Callback<CatResultModel> callback);
```

```
publishedDate: "Sat, 01 Oct 2015 15:11:15  
signedRedirectUrl: "http://news.google.com/  
O-XuV_Uu0ZVVw0QA",  
language: "ja",  
- image: {  
    url: "http://www.mylohas.net/images  
    tbUrl: "http://t3.gstatic.com/images  
    originalContextUrl: "http://www.mylo  
    publisher: "取材（一次情報メディア）（登  
    tbWidth: 80,  
    tbHeight: 43  
    }  
    },  
    - {  
        GsearchResultClass: "GnewsSearch",  
        clusterUrl: "http://news.google.com/news/s
```

```
[  
  - {
```

qiita api

body: "<p>こんにちは。エウセルソフトの田淵です。</p>
</i>Now On Tap って? </h3> <blockquote>
</blockquote> <p>Android 6 Marshmallow の(個人的
りかたに、ドキュメントが読める。という
ことは、全部ホームボタンが教えてくれる。「Google Ne
ってみる日本語版はまだか" class="fragment">
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3
<i class="fa fa-link"></i>使ってみる(日本語版はま
</p> <h5>
Tap はアプリ起動中にホームボタンを長押しするだけです
href="http://cdn-ak.f.st-hatena.com/images/fotolife/
hatena.com/images/fotolife/y/ytabuchi/20151006/
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3
ているところで Now on Tap ! "Xamarin Test Cloud rev
hatena.com/images/fotolife/y/ytabuchi/20151006/
hatena.com/images/fotolife/y/ytabuchi/20151006/

配列から始まる

```
public interface QiitaService {  
  
    @NonNull  
    @GET("/api/v1/tags/{tagName}/items")  
    void tagArticles(@Path("tagName") String tagName,  
                    Callback<List<QiitaModel>> callback);  
}
```

てきますw </p> <p><a href="http://cdn-ak.f.st-hate
hatena.com/images/fotolife/y/ytabuchi/20151006/
string の情報を引っ張って解析しているはず)、アプリ側
さそうですね。</p> <p>以上です。</p> ",
comment_count: 0,
created_at_as_seconds: 1444141459,
created_at_in_words: "約1時間",
created_at: "2015-10-06 23:24:19 +0900",


```
- responseData: {  
  - results: [  
    - {
```

web search api

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "",  
content: "2015年10月16日（金）東京・千駄  
10種類以上のフレーバーをたのしめる<b>アイ  
unescapeUrl: "http://www.mylohas.net/20  
url: "http%3A%2F%2Fwww.mylohas.net%2F  
title: "日本初上陸。乳製品も大豆も使わない「  
titleNoFormatting: "日本初上陸。乳製品も大豆  
location: "",  
publisher: "取材（一次情報メディア）（登録）",  
publisherDate: "Sun 04 Oct 2015 15:41:15  
signedRedirectUrl: "http://news.google.com/  
O-XuV_Uu0ZVVw0QA",  
language: "ja",  
- image: {  
  url: "http://www.mylohas.net/images/2  
tbUrl: "http://t3.gstatic.com/images?q:  
originalContextUrl: "http://www.myloha  
publisher: "取材（一次情報メディア）（登  
tbWidth: 80,  
tbHeight: 43  
}
```

配列を持ってる

```
- {
```

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "http://news.google.com/news/s
```

```
- {
```

qiita api

```
body: "<p>こんにちは。エグゼクティブの田淵です。</p>  
</i></a>Now On Tap って？</h3> <blockquote>  
</blockquote> <p>Android 6 Marshmallow の(個人的  
りから離れずにカードビューで表示してくれる。という  
ことは、全部ホームボタンが教えてくれる。「Google Ne  
ってみる日本語版はまだか" class="fragment"></span>  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
<i class="fa fa-link"></i></a>使ってみる(日本語版はま  
</p> <h5> <span id="初回実行" class="fragment"></span>  
Tap はアプリ起動中にホームボタンを長押しするだけで  
href="http://cdn-ak.f.st-hatena.com/images/fotolife/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
ているところで Now on Tap ! "Xamarin Test Cloud rev  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
Blog の検索と、Website へのリンクが表示されると共に  
いるようです。</p> <p><a href="http://cdn-ak.f.st-h  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
(株)ナビゲート という Web制作会社が出てきます！(ど  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
てきますw </p> <p><a href="http://cdn-ak.f.st-hate  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
string の情報を引っ張って解析しているはず)、アプリ側  
さそうですね。</p> <p>以上です。</p> ",  
comment_count: 0,  
created_at_as_seconds: 1444141459,  
created_at_in_words: "約1時間",  
created_at: "2015-10-06 23:24:19 +0900",
```



```
- responseData: {
```

```
- results: [  
  - {
```

web search api

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "",
```

```
public class CatResultModel {  
    @SerializedName("results")  
    List<CatNewsModel> mCatNewsModels;  
  
    public List<CatNewsModel> getCatNewsModels()  
    {  
        return mCatNewsModels;  
    }  
}
```

配列を持ってる

```
location: "",  
publisher: "取材（一次情報メディア）（登録）",  
publishDate: "Sun, 04 Oct 2015 15:41:15  
signedRedirectUrl: "http://news.google.com/  
O-XuV_Uu0ZVVw0QA",  
language: "ja",
```

```
public class CatNewsModel {  
    @SerializedName("image")  
    public CatImageModel mCatImageModel;  
  
    @SerializedName("title")  
    public String mTitle;  
  
    @SerializedName("unescape")  
    public String mUrl;  
  
    @SerializedName("published")  
    public Date mPublishDate;  
}
```

```
- {
```

qiita api

```
body: "<p>こんにちは。エクセルソフトの田淵です。</p>  
<h3>Now On Tap って？</h3> <blockquote>  
</blockquote> <p>Android 6 Marshmallow の(個人的な  
りから離れずにカードビューで表示してくれる。という
```

```
public class QiitaModel {  
    @SerializedName("user")  
    public QiitaUser mUser;  
  
    @SerializedName("title")  
    public String mTitle;  
  
    public class QiitaUser {  
        @SerializedName("url_name")  
        public String mName;  
        @SerializedName("profile_image_url")  
        public String mImageUrl;  
    }  
}
```

```
<p>以上です。</p> ",  
comment_count: 0,  
created_at_as_seconds: 1444141459,  
created_at_in_words: "約1時間",  
created_at: "2015-10-06 23:24:19 +0900",
```



```
- responseData: {  
  - results: [  
    - {
```

web search api

indexがっている

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "",  
content: "2015年10月16日（金）東京・千駄  
10種類以上のフレーバーをたのしめる<b>アイ  
unescapeUrl: "http://www.mylohas.net/20  
url: "http%3A%2F%2Fwww.mylohas.net%2F  
title: "日本初上陸。乳製品も大豆も使わない「  
titleNoFormatting: "日本初上陸。乳製品も大豆  
location: "",  
publisher: "取材（一次情報メディア）（登録）",  
publishedDate: "Sun, 04 Oct 2015 15:41:15  
signedRedirectUrl: "http://news.google.com/  
O-XuV_Uu0ZVVw0QA",  
language: "ja",  
- image: {  
  url: "http://www.mylohas.net/images/2  
tbUrl: "http://t3.gstatic.com/images?q:  
originalContextUrl: "http://www.myloha  
publisher: "取材（一次情報メディア）（登  
tbWidth: 80,  
tbHeight: 43  
}
```

```
},  
- {
```

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "http://news.google.com/news/s
```

```
- {
```

qiita api

```
body: "<p>こんにちは。エウセルソフトの田淵です。</p>  
</i></a>Now On Tap って？</h3> <blockquote>  
</blockquote> <p>Android 6 Marshmallow の(個人的  
りから離れずにカードビューで表示してくれる。という  
ことは、全部ホームボタンが教えてくれる。「Google Ne  
ってみる日本語版はまだか" class="fragment"></span>  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
<i class="fa fa-link"></i></a>使ってみる(日本語版はま  
</p> <h5> <span id="初回実行" class="fragment"></span>  
Tap はアプリ起動中にホームボタンを長押しするだけで  
href="http://cdn-ak.f.st-hatena.com/images/fotolife/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
href="#%E4%BD%BF%E3%81%A3%E3%81%A6%E3  
ているところで Now on Tap ! "Xamarin Test Cloud rev  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
Blog の検索と、Website へのリンクが表示されると共に  
いるようです。</p> <p><a href="http://cdn-ak.f.st-h  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
(株)ナビゲート という Web制作会社が出てきます！(ど  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
てきますw </p> <p><a href="http://cdn-ak.f.st-hate  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
string の情報を引っ張って解析しているはず)、アプリ側  
さそうですね。</p> <p>以上です。</p> ",  
comment_count: 0,  
created_at_as_seconds: 1444141459,  
created_at_in_words: "約1時間",  
created_at: "2015-10-06 23:24:19 +0900",
```



```
- responseData: {  
  - results: [  
    - {
```

web search api

indexがつかっている

```
GsearchResultClass: "GnewsSearch",  
clusterUrl: "",  
content: "2015年10月16日 (金) 東京・千駄  
10種類以上のフレーバーをたのしめる<b>アイ  
unescapeUrl: "http://www.mylohas.net/20  
url: "http%3A%2F%2Fwww.mylohas.net%2  
title: "日本初上陸。乳製品も大豆も使わない「  
titleNoFormatting: "日本初上陸。乳製品も大豆  
location: "",  
publisher: "取材（一次情報メディア）（登録）"  
publishedDate: "Sun, 04 Oct 2015 15:41:15  
signedRedirectUrl: "http://news.google.com/
```

qiita api

```
body: "<p>こんにちは。エクセルソフトの田淵です。</p>  
</i></a>Now On Tap って？</h3> <blockquote>  
</blockquote> <p>Android 6 Marshmallow の(個人的  
りから離れずにカードビューで表示してくれる。という  
ことは、全部ホームボタンが教えてくれる。「Google N
```

```
public class QiitaApiClient {  
  
    private final static String QIITA_API_HOST = "https://qiita.com/";  
  
    public static void getModels(String tagName,  
                                Callback<List<QiitaModel>> callback) {  
  
        RestAdapter restAdapter = new RestAdapter.Builder()  
            .setEndpoint(QIITA_API_HOST)  
            .build();  
  
        QiitaService qiitaService =  
            restAdapter.create(QiitaService.class);  
        qiitaService.getModels(tagName, callback);  
    }  
}
```

```
public class CatNewsClient {  
  
    private final static String CAT_NEWS_API_HOST =  
        "https://ajax.googleapis.com/";  
  
    public static void getResultModel(String keyword,  
                                     Callback<CatResultModel> callback) {  
  
        Gson gson = new GsonBuilder()  
            .registerTypeAdapter(CatResultModel.class, new CatDeserializer())  
            .create();  
  
        RestAdapter restAdapter = new RestAdapter.Builder()  
            .setEndpoint(CAT_NEWS_API_HOST)  
            .setConverter(new GsonConverter(gson))  
            .build();  
  
        CatNewsService catNewsService =  
            restAdapter.create(CatNewsService.class);  
        catNewsService.results(keyword, callback);  
    }  
}
```

```
hatena.com/images/fotolife/y/ytabuchi/20151006/  
Blog の検索と、Website へのリンクが表示されると共に  
いるようです。</p> <p><a href="http://cdn-ak.f.st-h  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
(株)ナビゲート という Web制作会社が出てきます！（ど  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
てきますw </p> <p><a href="http://cdn-ak.f.st-hate  
hatena.com/images/fotolife/y/ytabuchi/20151006/  
string の情報を引っ張って解析しているはず）、アプリ側  
さそうですね。</p> <p>以上です。</p> ",  
comment_count: 0,  
created_at_as_seconds: 1444141459,  
created_at_in_words: "約1時間",  
created_at: "2015-10-06 23:24:19 +0900",
```

CATDESERIALIZER

- JsonSerializerを継承
- Gsonのパースをカスタマイズ

```
public class CatDeserializer implements JsonSerializer {  
    @Override  
    public CatResultModel deserialize(JsonElement json, Type typeOfT,  
                                     JsonSerializerContext context) throws JsonParseException {  
        Log.d(MyApplication.TAG, "deserialize: " + json.toString());  
        JsonElement content = json.getAsJsonObject().get("responseData");  
  
        // Wed, 30 Sep 2015 23:56:15 -0700  
        Gson gson = new GsonBuilder()  
            .setDateFormat("EEE, dd MMM yyyy HH:mm:ss z")  
            .create();  
        return gson.fromJson(content, typeOfT);  
    }  
}
```


CATDESERIALIZER

- jsonのパース位置を調整
- Dateオブジェクトのformatを設定

```
public class CatDeserializer implements JsonDeserializer {  
  
    @Override  
    public CatResultModel deserialize(JsonElement json, Type typeOfT,  
                                     JsonDeserializationContext context) throws JsonParseException {  
        Log.d(MvApplication.TAG, "deserialize: " + json.toString());  
        JsonElement content = json.getAsJsonObject().get("responseData");  
  
        //      Wed, 30 Sep 2015 23:56:15 -0700  
        Gson gson = new GsonBuilder()  
            .setDateFormat("EEE, dd MMM yyyy HH:mm:ss z")  
            .create();  
        return gson.fromJson(content, typeOfT);  
    }  
}
```

- CatResultModelを受け取り、メンバーの
List<CatNewsModel>から取り出す

```
private void fetchGoogleSearchApi() {
    CatNewsClient.getResultModel("猫", new Callback<CatResultModel>() {
        @Override
        public void success(CatResultModel catResultModel, Response response) {
            Toast.makeText(MainActivity.this, "success", Toast.LENGTH_SHORT).show();
            List<CatNewsModel> catNewsModels = catResultModel.getCatNewsModels();
            Log.d(MyApplication.TAG, "size: " + catNewsModels.size());
            for (CatNewsModel catNewsModel : catNewsModels) {
                Log.d(MyApplication.TAG, catNewsModel.getTitle());
                Log.d(MyApplication.TAG, catNewsModel.getCatImageUrl());
            }
        }

        @Override
        public void failure(RetrofitError error) {
            Toast.makeText(MainActivity.this, error.getMessage(), Toast.LENGTH_SHORT).show();
            Log.d(MyApplication.TAG, "failure url : " + error.getUrl());
            Log.d(MyApplication.TAG, "failure message : " + error.getMessage());
        }
    });
}
```


効果測定

効果測定

- 実際のデータを見て、施策の効果を測る。
- 効果を測定するためにデータを取る
 - Google Analytics

GOOGLE ANALYTICS

- GAとよく略される
- 無料で使え、開発も活発。
- 個人を特定するデータを送ってはいけない

導入(V4)

- Google Developer Consoleに
プロジェクトの追加が必要
- 以下から行けばチュートリアルに則り
作成できる
 - [https://developers.google.com/mobile/add?
platform=android&cntapi=analytics](https://developers.google.com/mobile/add?platform=android&cntapi=analytics)
(<https://goo.gl/5YZ7D2>)

- projectルートのbuild.gradle
- app/build.gradle
- configuration fileの追加
- AndroidManifest.xml
- Applicationクラスでの初期化

```
buildscript {  
    repositories {  
        jcenter()  
    }  
    dependencies {  
        classpath 'com.android.tools.build:gradle:1.3.0'  
        classpath 'com.google.gms:google-services:1.4.0-beta3'  
    }  
}  
  
allprojects {  
    repositories {  
        jcenter()  
    }  
}
```

- projectルートのbuild.gradle


```
apply plugin: 'com.android.application'
apply plugin: 'com.google.gms.google-services'

android {
    compileSdkVersion 23
    buildToolsVersion '23.0.1'

    defaultConfig {
        applicationId "com.sakebook.android.trunksimplenews"
        minSdkVersion 16
        targetSdkVersion 23
        versionCode 1
        versionName "1.0"
        multiDexEnabled true
    }
    buildTypes {
        // ...
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    compile 'com.android.support:appcompat-v7:23.0.1'
    compile 'com.squareup.retrofit:retrofit:1.9.0'
    compile 'com.squareup.picasso:picasso:2.5.2'
    compile 'com.google.android.gms:play-services-analytics:8.1.0'
}
```

- app/build.gradle

- チュートリアルで作成した
「google-services.json」 を app/ に配置


```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.sakebook.android.trunksimplenews" >

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />

    <application
        android:name=".TrunksimpleNewsApplication"
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >

        <!-- Add the following meta-data for devices running Google Play services -->
        <meta-data
            android:name="com.google.android.gms.version"
            android:value="@integer/google_play_services_version" />

        <activity
            android:name=".activities.MainActivity"
            android:label="@string/app_name" >

```

- AndroidManifest.xml

APPLICATION

- アプリ起動時と終了時に一度だけ呼ばれるクラス
- 初回起動のActivityよりも先に呼ばれる
 - Applicationを継承したクラスを作成する


```

public class TrunksimpleNewsApplication extends Application {

    private Tracker mTracker;

    @Override
    public void onCreate() {
        super.onCreate();
        L.d("TrunkSimpleNewsApplication onCreate");
    }

    synchronized public Tracker getDefaultTracker() {
        if (mTracker == null) {
            GoogleAnalytics analytics = GoogleAnalytics.getInstance(this);
            // To enable debug logging use: adb shell setprop log.tag.GAv4 DEBUG
            mTracker = analytics.newTracker(R.xml.global_tracker);
            analytics.setDryRun(true);
        }
        return mTracker;
    }
}

```

- trackerのインスタンス取得メソッド
- Applicationクラスは終了時まで生き続ける

準備完了

- 主に計測するのは次の2種類
 - スクリーン
 - 画面が表示された時など
 - イベント
 - それ以外の任意の状況

スクリーン

```
TrunkSimpleNewsApplication application =  
    (TrunkSimpleNewsApplication) getApplication();  
Tracker t = application.getDefaultTracker();  
t.setScreenName(this.getClass().getSimpleName());  
t.send(new HitBuilders.ScreenViewBuilder().build());
```

アクティブ スクリーン名	
1.	MainActivity

- BaseActivityを作成し onCreate や onStart など
計測するようにするとまとめられて楽
- 最近のGAだと自動で計測できる方法もある
 - Tracker.enableAutoActivityTracking(enabled)
 - xml要作成
 - <http://qiita.com/takashibagura/items/79a560c739b3a8af2714>

イベント

```
TrunkSimpleNewsApplication application =  
    (TrunkSimpleNewsApplication)application();  
Tracker t = application.getDefaultTracker();  
t.send(new HitBuilders.EventBuilder()  
    .setCategory("UX")  
    .setAction("click")  
    .setLabel("license")  
    .build());
```


イベント

現在

1

人のアクティブ ユーザーがアプリを利用しています

■ 携帯電話



イベント

分単位

1.5

1.0

0.5

-26 分

-21 分

-16 分

-11 分

表示: [アクティブ ユーザー](#) イベント (直前の 30 分間)

指標の合計: 1

イベント アクション

イベント ラベル

1. [click](#)

[license](#)

イベント カテゴリ

イベント アクション

1. [UX](#)

[click](#)

APK署名

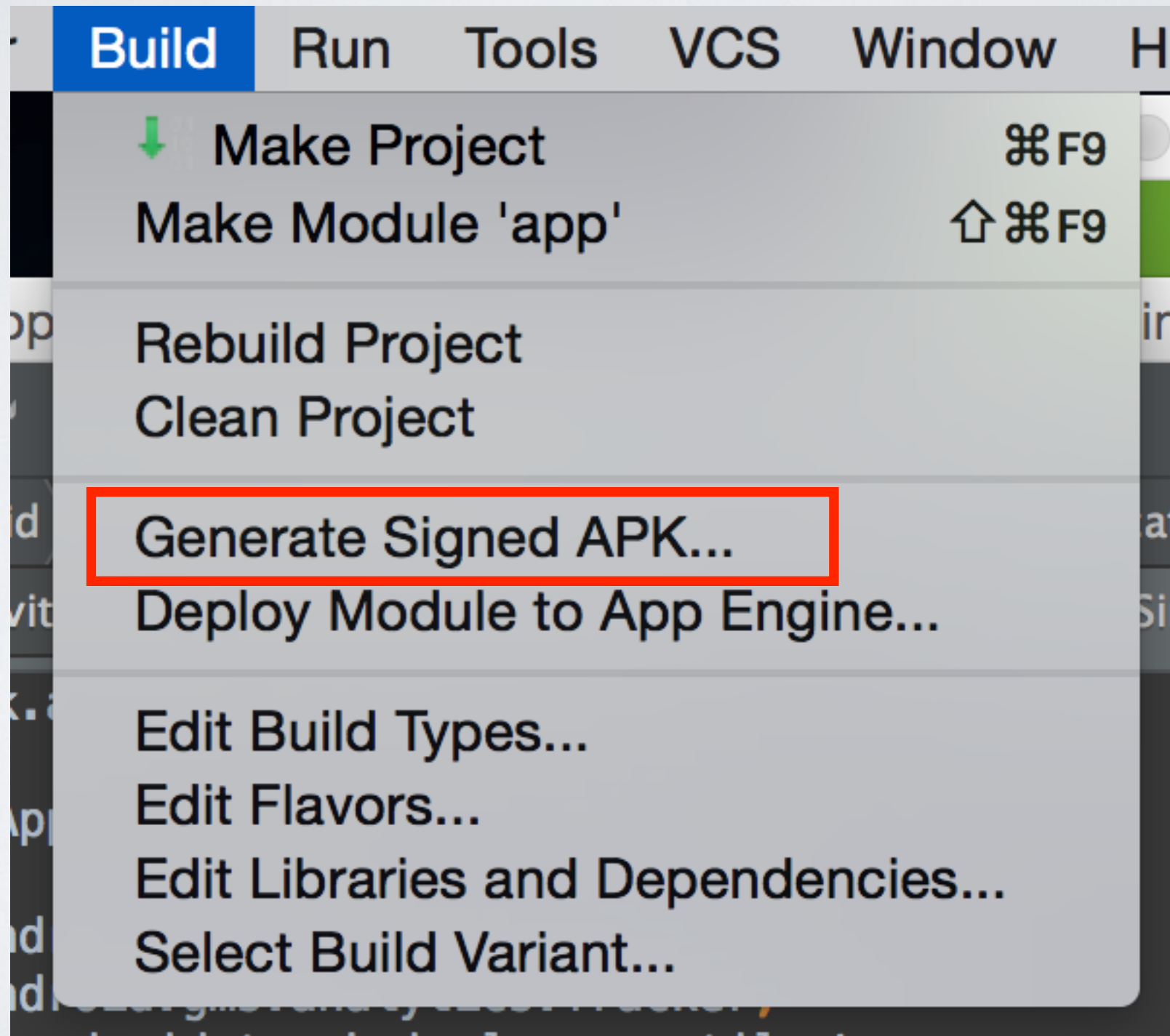
APK署名

- 署名されたAPK以外は端末にインストールできない
- 署名にはReleaseとDebugがある
 - 開発中はDebugで署名されている

APK署名

- すでに端末に入っているAPKと
同じパッケージネームでも署名が異なると
上書きできない
- 同一署名がされたAPK同士は
データのやり取りが可能
- 署名をなくすと同じものは復元できない

署名(証明書)作成



- .jksファイル
- keystoreのパスワード
- Aliasのパスワード
- Certificateは必須ではない

New Key Store

Key store path: ...

Password: Confirm:

Key _____

Alias:

Password: Confirm:

Validity (years): ▴ ▾

Certificate

First and Last Name:

Organizational Unit:

Organization:





















































City or Locality:

State or Province:

Country Code (XX):

Cancel OK

GENYMOTIONで動かそう

 .DS_Store			 .gitignore	
 .gitignore			 app-release.apk	
 .gradle			 app.iml	
 .idea			 build	
 app			 build.gradle	
 build			 google-services.json	
 build.gradle			 libs	
 gradle			 proguard-rules.pro	
 gradle.properties			 src	
 gradlew			 trunk.jks	
 gradlew.bat				
 local.properties				
 settings.gradle				
 TrunkSimpleNews.iml				

LOG

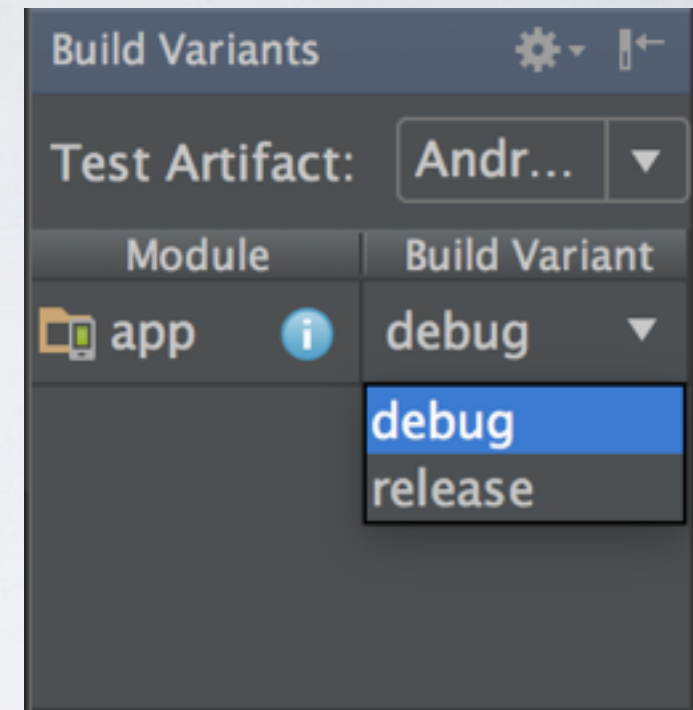
- 相変わらずLogが見える
- 情報漏洩に繋がるので、見えないようにする
 - 手動で消していく？

期待値

- コードを人力で変更したくない
 - バグの元
- 開発版だけログが出て、本番はログが出ない
- アプリの状態を判定できるのか？
 - Build Typesで判定できる

BUILD TYPES

- 名の通りビルドの種類
- debugとreleaseは
標準で用意されている
- 追加することも可能




```
defaultConfig {
    applicationId "com.sakebo
    minSdkVersion 16
    targetSdkVersion 23
    versionCode 1
    versionName "1.0"
    multiDexEnabled true
}

buildTypes {
    release {
        minifyEnabled false
        proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
    }
}
```

判定

- 「BuildConfig」 というクラスが
自動生成されているのでそれを用いる
- パッケージネームやバージョンもわかる

```
package com.sakebook.android.trunksimplenews;  
  
public final class BuildConfig {  
    public static final boolean DEBUG = Boolean.parseBoolean("true");  
    public static final String APPLICATION_ID = "com.sakebook.android.trunksimplenews";  
    public static final String BUILD_TYPE = "debug";  
    public static final String FLAVOR = "";  
    public static final int VERSION_CODE = 1;  
    public static final String VERSION_NAME = "1.0";  
}
```

LOGを加工

- Logクラスをラップする
- BuildTypesがdebugのときはLogが呼ばれない

- 表示されない！

```
public class L {  
  
    private final static String TAG =  
        TrunkSimpleNewsApplication.cl  
  
    public static void d(String msg) {  
        if (BuildConfig.DEBUG) {  
            Log.d(TAG, msg);  
        }  
    }  
}
```


リバーエンジニアリング

- .apkは.zipに置き換えることができる。
- 中身を展開して、情報を抜き取ったり
悪意のある改ざんを行い、
パッケージングされる可能性がある
- 無防備でいるわけにはいかない！

難読化

難読化

- ソース(java)を一文字で表現するなどして、コード量の圧縮と悪意のある人から読み解かれにくくする。

PROGUARD

- Gradleに対応している難読化のためのツール
- 難読化対象・及び非対象、内容を記述する
- 通常難読化はrelease buildのみに行う

PROGUARDの設定

- minifyEnabled: 難読化を行うかどうか
- proguardFiles: 難読化のルールを指定

```
buildTypes {  
    debug {  
        minifyEnabled false  
    }  
  
    release {  
        minifyEnabled true  
        proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'  
    }  
}
```

PROGUARD-RULES.PRO

- proguard-android.txt
 - AndroidのSDK周りで必要な記述がされている
- proguard-rules.pro
 - 追加で必要な記述をここに書く

どう書けば？

- 自身のコードはそのまま難読化にかけても動くかもしれないが
果たしてライブラリはそうだろうか？
 - 主にライブラリを難読化対象から外す
内容を記述することになる

- 追加で使ったライブラリ
 - retrofit
 - Picasso
 - PlayServicesAnalytics
- 公式ページに大体書いてある

Retrofit 1.X

-keep class com.squareup.okhttp.** { *; }

-keep class retrofit.** { *; }

-keep interface com.squareup.okhttp.** { *; }

-dontwarn com.squareup.okhttp.**

-dontwarn okio.**

-dontwarn retrofit.**

-dontwarn rx.**

-keepclasseswithmembers class * {

 @retrofit.http.* <methods>;

}

If in your rest service interface you use methods with Callback argument.

-keepattributes Exceptions

If your rest service methods throw custom exceptions, because you've defined an ErrorHandler.

-keepattributes Signature

google play services

-keep public class com.google.android.gms.analytics.** {
 public *;

}

AppCompatActivity

-keep class android.support.v7.app.** { *; }

-keep class android.support.v4.app.** { *; }

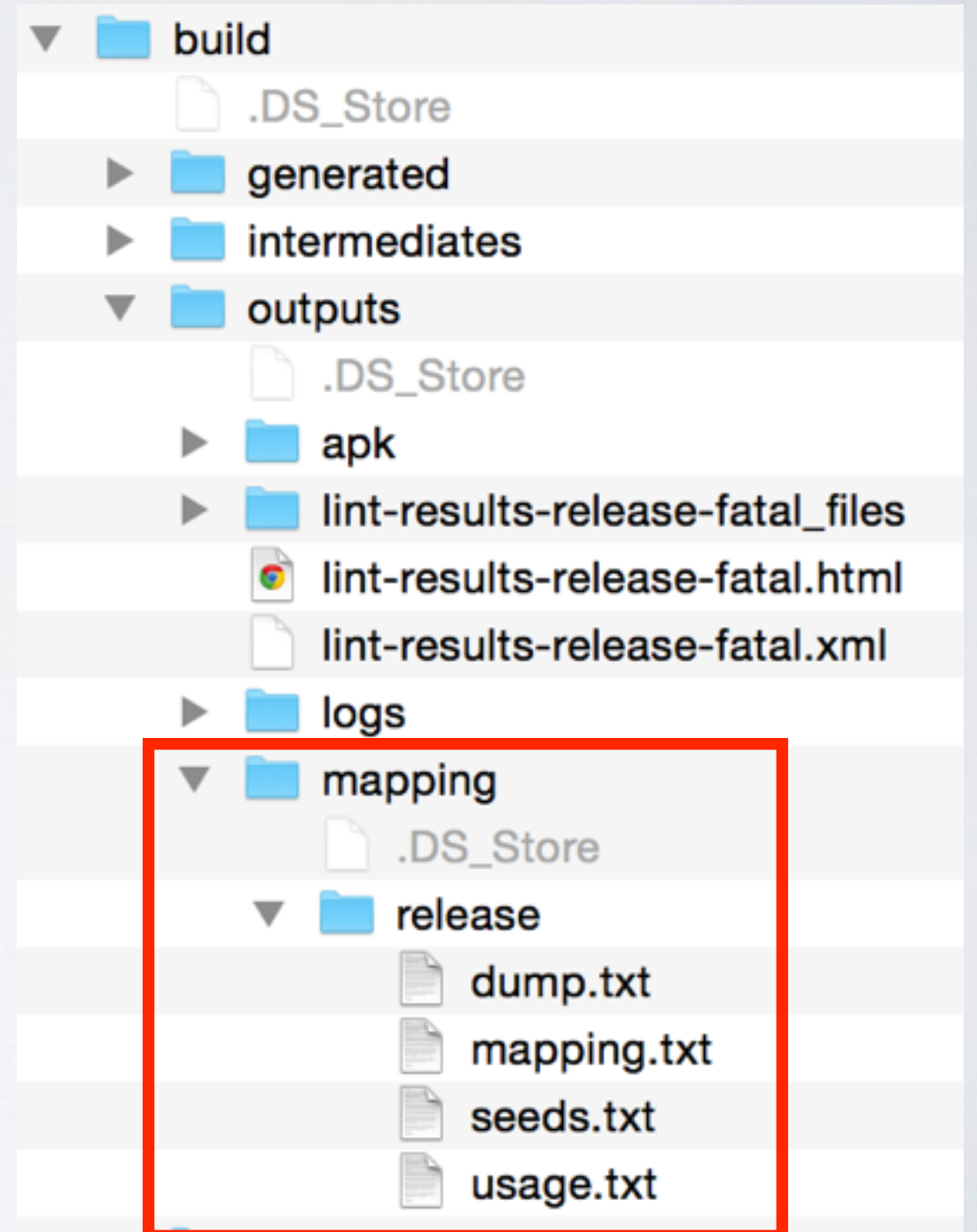
-keep interface android.support.v4.app.** { *; }

#picasso

-dontwarn com.squareup.okhttp.**

誰も読めないんじゃない？

- mappingが生成されるので、これを使い戻す。
- ReleaseAPKとセットで保存しておく



- `${sdk}/tools/proguard/bin/proguardgui.sh`
- エラー内容(スタックトレース)

- `mapping.txt`

