## **开篇**

一如前端深似海，从此节操是路人，从此再无安宁日，从此红尘是路人。要说技术更迭速度，还有比前端更快的么，根本停不下来。

这不，Google刚发布Flutter不到一年时间，1.0正式版发布不到两个月。

阿里系的闲鱼老大哥，已经率先用Flutter重构了闲鱼，虽然没完全重构，但高频的重度页面都是Flutter的了。

这一幕似曾相识，当初RN出来的时候不也是闲鱼团队先吃的螃蟹吗，在这里向闲鱼团队的老哥们致敬:hatching\_chick:。

既然老大哥都出动了，也侧面验证了这项技术的可行性。当小弟的也不能落后嘛，每天抽时间断断续续的学了两周时间，仿部分知乎的客户端，撸了一套客户端出来。

前一周主要是熟悉Dart语言和常规的客户端布局方式，后一周主要是掌握使用HTTP的请求、下拉上拉、左滑右滑、长按等常用手势、相机调用、video播放等进阶用法。

两周下来，基本上可以开发80%以上常见的客户端需求。

前期一直在用simulator开发，略有卡顿，心中难免有些疑惑。结果最后release打包到手机后，竟然如丝般顺滑！！！简直喜出望外，完全可以睥睨原生开发，在这一点上的确要优于目前的RN

最重要的是作为Materail Design极简又有质感风格的鸭狗血粉丝，Flutter造出来的界面简直倍爽。至此正式入坑Flutter开发。Google万岁！

Flutter是谷歌的移动UI框架，可以快速在iOS和Android上构建高质量的原生用户界面。 Flutter可以与现有的代码一起工作。在全世界，Flutter正在被越来越多的开发者和组织使用，并且Flutter是完全免费、开源的。

这里把学习过程中一些常用高频的东西总结出来，基本能满足大多数情况下的开发需求。

## **Scaffold 主要的属性说明**

* appBar：显示在界面顶部的一个 AppBar
* body：当前界面所显示的主要内容
* floatingActionButton： 在 Material 中定义的一个功能按钮。
* persistentFooterButtons：固定在下方显示的按钮。 [https://material.google.com/c...](https://link.zhihu.com/?target=https://material.google.com/components/buttons.html%23buttons-persistent-footer-buttons" \t "https://zhuanlan.zhihu.com/p/_blank)
* drawer：侧边栏控件
* bottomNavigationBar：显示在底部的导航栏按钮栏。可以查看文档：Flutter学习之制作底部菜单导航
* backgroundColor：背景颜色
* resizeToAvoidBottomPadding： 控制界面内容 body 是否重新布局来避免底部被覆盖了，比如当键盘显示的时候，重新布局避免被键盘盖住内容。默认值为 true。

## **底部菜单 bottomNavigationBar，Tab栏切换 TabBar**

TabController controller;

@override

void initState() {

super.initState();

// initialize the tab controller

// vsync ??

controller = new TabController(length: 5, vsync: this);

}

@override

void dispose() {

// dispose of tab controller

controller.dispose();

super.dispose();

}

...

body: new TabBarView(

children: <Widget>[new HomeTab(), new IdeaTab(), new ColleagueTab(), new MessageTab(), new MeTab()],

controller: controller,

),

bottomNavigationBar: new Material(

// background color of bottom navigation bar

color: Colors.white,

textStyle: new TextStyle(

color: Colors.black45

),

child: new TabBar(

unselectedLabelColor: Colors.black45,

labelColor: Colors.blue,

controller: controller,

tabs: <Tab>[

new Tab(

child: new Container(

padding: EdgeInsets.only(top: 5),

child: new Column(

children: <Widget>[

Icon(Icons.home, size: 25,),

Text('首页', style: TextStyle(fontSize: 10),)

],

),

),

),

new Tab(

child: new Container(

padding: EdgeInsets.only(top: 5),

child: new Column(

children: <Widget>[

Icon(Icons.access\_alarm, size: 25,),

Text('想法', style: TextStyle(fontSize: 10),)

],

),

),

),

new Tab(

child: new Container(

padding: EdgeInsets.only(top: 5),

child: new Column(

children: <Widget>[

Icon(Icons.access\_time, size: 25,),

Text('大学', style: TextStyle(fontSize: 10),)

],

),

),

),

new Tab(

child: new Container(

padding: EdgeInsets.only(top: 5),

child: new Column(

children: <Widget>[

Icon(Icons.account\_balance\_wallet, size: 25,),

Text('消息', style: TextStyle(fontSize: 10),)

],

),

),

),

new Tab(

child: new Container(

padding: EdgeInsets.only(top: 5),

child: new Column(

children: <Widget>[

Icon(Icons.adb, size: 25,),

Text('我的', style: TextStyle(fontSize: 10),)

],

),

),

),

],

),

),

效果：



## **顶栏自定义 appbar：title属性 顶部搜索栏**

@override

Widget build(BuildContext context) {

return new Scaffold(

appBar: new AppBar(

title: searchBar(),

backgroundColor: Colors.white,

bottom: new Text('bottom'),

),

body: new Container()

);

}

/\*\*

\* 顶部搜索栏

\*/

Widget searchBar() {

return new Container(

child: new Row(

children: <Widget>[

new Expanded(

child: new FlatButton.icon(

color:Color.fromRGBO(229, 229, 229, 1.0),

onPressed: (){

Navigator.of(context).push(new MaterialPageRoute(builder: (context){

return new SearchPage();

}));

},

icon: new Icon(

Icons.search,

color: Colors.black38,

size: 16.0,

),

label: new Text(

"诺奖得主为上课推迟发布会",

style: new TextStyle(color: Colors.black38)

),

),

),

new Container(

child: new FlatButton.icon(

onPressed: (){

Navigator.of(context).push(new MaterialPageRoute(builder: (context){

return new AskPage();

}));

},

icon: new Icon(

Icons.border\_color,

color: Colors.blue,

size: 14.0

),

label: new Text(

'提问',

style: new TextStyle(color: Colors.blue),

),

),

)

],

),

);

}



## **图片圆角**

Container(

margin: EdgeInsets.only(right: 5),

decoration: new BoxDecoration(

shape: BoxShape.circle,

image: new DecorationImage(

image: new NetworkImage(avatarUrl),

)

),

width: 30,

height: 30,

),

## **数组里动态添加组件**

bool notNull(Object o) => o != null;

Widget build() {

return new Column(

children: <Widget>[

new Title(),

new Body(),

shouldShowFooter ? new Footer() : null

].where(notNull).toList(),

);

}

## **Text显示指定行数，超出后显示省略号**

Text(

content,

maxLines: 3,

overflow: TextOverflow.ellipsis,

style: new TextStyle(fontSize: 14, color: Colors.black54),

),

## **margin 负值**

[https://stackoverflow.com/que...](https://link.zhihu.com/?target=https://stackoverflow.com/questions/42257668/the-equivalent-of-wrap-content-and-match-parent-in-flutter" \t "https://zhuanlan.zhihu.com/p/_blank)

return Container(

width: 40,

height:40,

// flutter中的margin没有负值的说法

// https://stackoverflow.com/questions/42257668/the-equivalent-of-wrap-content-and-match-parent-in-flutter

transform: Matrix4.translationValues(-20.0, 0.0, 0.0),

decoration: new BoxDecoration(

border: Border.all(width: 3, color: Colors.white),

color: Colors.black,

shape: BoxShape.circle,

image: new DecorationImage(

image: new NetworkImage('https://pic3.zhimg.com/50/d2af1b6b1\_s.jpg')

)

),

);

## **图片自适应填满container**

[https://stackoverflow.com/que...](https://link.zhihu.com/?target=https://stackoverflow.com/questions/45745448/how-do-i-stretch-an-image-to-fit-the-whole-background-100-height-x-100-width" \t "https://zhuanlan.zhihu.com/p/_blank)

new Container(

height: 200,

decoration: new BoxDecoration(

image: new DecorationImage(

image: NetworkImage('https://pic3.zhimg.com/50/v2-f9fd4b13a46f2800a7049a5724e5969f\_400x224.jpg'),

fit: BoxFit.fill

)

),

),

## **布局方式**

justify-content: mainAxisAlignment

align-items: crossAxisAlignment

column 设置crossAxisAlignment: stretch后子元素宽度为100%，如果想让子元素宽度不为100%， 将其包裹在Row元素中即可。

flutter row and column

[https://medium.com/jlouage/fl...](https://link.zhihu.com/?target=https://medium.com/jlouage/flutter-row-column-cheat-sheet-78c38d242041" \t "https://zhuanlan.zhihu.com/p/_blank)

## **捕捉点击事件**

使用GestureDetector包裹widget即可。

onTap: click,

child: Text(

name,

style: TextStyle(color: Colors.black87),

),

),

## **Dart 数组方法**

[https://codeburst.io/top-10-a...](https://link.zhihu.com/?target=https://codeburst.io/top-10-array-utility-methods-you-should-know-dart-feb2648ee3a2" \t "https://zhuanlan.zhihu.com/p/_blank)

## **PopupMenuButton 下拉弹窗菜单**

[https://stackoverflow.com/que...](https://link.zhihu.com/?target=https://stackoverflow.com/questions/43349013/how-to-open-a-popupmenubutton" \t "https://zhuanlan.zhihu.com/p/_blank)

@override

DetailPageState createState() => DetailPageState();

}

class DetailPageState extends State<DetailPage> {

final GlobalKey \_menuKey = new GlobalKey();

....

....

....

child: new Row(

children: <Widget>[

new Container(

child: new GestureDetector(

onTap: () {

dynamic state = \_menuKey.currentState;

state.showButtonMenu();

},

child: new Container(

child: new Text('默认排序'),

),

),

),

new PopupMenuButton(

icon: Icon(Icons.keyboard\_arrow\_down),

offset: Offset(0, 50),

key: \_menuKey,

itemBuilder: (\_) => <PopupMenuItem<String>>[

new PopupMenuItem<String>(

child: const Text('默认排序'), value: 'default'),

new PopupMenuItem<String>(

child: const Text('按时间排序'), value: 'timeline'),

],

onSelected: (\_) {}

)

],

),



## **分割线**

水平分割线 Divider

垂直分割线 VerticalDivider (无效？？？)

## **swiper**

[https://pub.dartlang.org/pack...](https://link.zhihu.com/?target=https://pub.dartlang.org/packages/flutter_swiper" \t "https://zhuanlan.zhihu.com/p/_blank)

...

var images = [

'https://pic3.zhimg.com/v2-5806d9e33e36fa772c8da56c931bb416\_b.jpg',

'https://pic1.zhimg.com/50/v2-f355ca177e011626938b479f0e2e3e03\_hd.jpg',

'https://pic2.zhimg.com/v2-d8e47ed961b93b875ad814104016bdfd\_b.jpg'

];

child: new Swiper(

itemBuilder: (BuildContext context,int index){

return new Image.network(images[index], fit: BoxFit.cover,);

},

itemCount: 3,

pagination: new SwiperPagination(),

//control: new SwiperControl(),

),

## **floatingActionButton 浮动button**

[https://proandroiddev.com/a-d...](https://link.zhihu.com/?target=https://proandroiddev.com/a-deep-dive-into-floatingactionbutton-in-flutter-bf95bee11627" \t "https://zhuanlan.zhihu.com/p/_blank)

floatingActionButton 配合 Scaffold 使用最佳

Scaffold(

floatingActionButton: new FloatingActionButton(

onPressed: (){},

child: Icon(Icons.edit),

//mini: true,

),

// 默认右下角，可设置位置。

floatingActionButtonLocation: FloatingActionButtonLocation.centerFloat,

)

## **滑动视图**

SingleChildScrollView

水平方向滑动 scrollDirection: Axis.horizontal

## **高斯模糊**

[https://stackoverflow.com/que...](https://link.zhihu.com/?target=https://stackoverflow.com/questions/43550853/how-do-i-do-the-frosted-glass-effect-in-flutter" \t "https://zhuanlan.zhihu.com/p/_blank)

import 'dart:ui';

new BackdropFilter(

filter: new ImageFilter.blur(sigmaX: 10.0, sigmaY: 10.0),

child: Text(desc, style: TextStyle(color: Colors.white),),

),

## **对话框弹窗**

[https://docs.flutter.io/flutt...](https://link.zhihu.com/?target=https://docs.flutter.io/flutter/material/AlertDialog-class.html" \t "https://zhuanlan.zhihu.com/p/_blank)

AlertDialog

// flutter defined function

showDialog(

context: context,

builder: (BuildContext context) {

// return object of type Dialog

return AlertDialog(

title: Text('Rewind and remember'),

content: SingleChildScrollView(

child: ListBody(

children: <Widget>[

Text('You will never be satisfied.'),

Text('You\’re like me. I’m never satisfied.'),

],

),

),

actions: <Widget>[

// usually buttons at the bottom of the dialog

new FlatButton(

child: new Text("Close"),

onPressed: () {

Navigator.of(context).pop();

},

),

],

);

},

);

}

// 调用

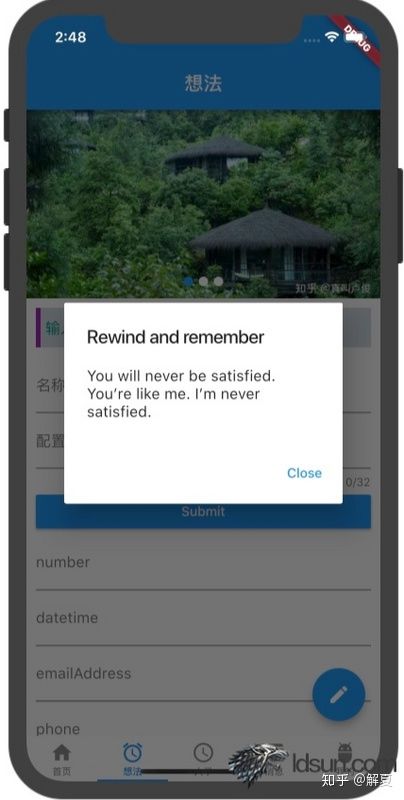
....

onPressed: (){

\_showDialog(context);

},

....



## **HTTP 请求、JSON编码解码**

[https://flutterchina.club/net...](https://link.zhihu.com/?target=https://flutterchina.club/networking/" \t "https://zhuanlan.zhihu.com/p/_blank)

// 加载库

import 'dart:convert';

import 'dart:io';

// 请求

try {

var request = await httpClient.getUrl(Uri.parse(url));

var response = await request.close();

if (response.statusCode == HttpStatus.OK) {

var json = await response.transform(UTF8.decoder).join();

var data = JSON.decode(json);

result = data['origin'];

} else {

result =

'Error getting IP address:\nHttp status ${response.statusCode}';

}

} catch (exception) {

result = 'Failed getting IP address';

}

// 保存返回的数据

// error: setState() called after dispose()

// If the widget was removed from the tree while the message was in flight,

// we want to discard the reply rather than calling setState to update our

// non-existent appearance.

if (!mounted) return;

setState(() {

\_ipAddress = result;

});

## **时间控制：延时**

import 'dart:async';

Future<Null> \_onRefresh() {

Completer<Null> completer = new Completer<Null>();

new Timer(new Duration(seconds: 3), () {

print("timer complete");

completer.complete();

});

return completer.future;

}

## **下拉刷新 RefreshIndicator**

new RefreshIndicator(

onRefresh: \_onRefresh,

child: new SingleChildScrollView(

child: new Container(

padding: EdgeInsets.all(10),

child: new Text(\_jsonData),

),

),

)

## **上拉加载更多**

[https://juejin.im/post/5b3abf...](https://link.zhihu.com/?target=https://juejin.im/post/5b3abfc4518825622c14a6f1" \t "https://zhuanlan.zhihu.com/p/_blank)

ScrollController \_controller = new ScrollController();

@override

void initState() {

super.initState();

\_controller.addListener((){

if(\_controller.position.pixels == \_controller.position.maxScrollExtent) {

print('下拉加载');

\_getMoreData();

}

});

}

@override

void dispose() {

\_controller.dispose();

super.dispose();

}

...

scroll controller: \_controller

...

## **flutter运行模式**

[https://www.jianshu.com/p/4db...](https://link.zhihu.com/?target=https://www.jianshu.com/p/4db65478aaa3" \t "https://zhuanlan.zhihu.com/p/_blank)

## **报错解决**

1、在安卓真机release后 ios simulator无法编译

Launching lib/main.dart on iPhone X in debug mode...

Xcode build done. 1.0s

Failed to build iOS app

Error output from Xcode build:

↳

\*\* BUILD FAILED \*\*

Xcode's output:

↳

=== BUILD TARGET Runner OF PROJECT Runner WITH CONFIGURATION Debug ===

diff: /Users/ludis/Desktop/opt/flutter/zh/ios/Pods/Manifest.lock: No such file or directory

error: The sandbox is not in sync with the Podfile.lock. Run 'pod install' or update your CocoaPods installation.

Could not build the application for the simulator.

Error launching application on iPhone X.

Exited (sigterm)

解决

cd ios

pod install

[https://medium.com/flutter-co...](https://link.zhihu.com/?target=https://medium.com/flutter-community/scrolling-animation-in-flutter-6a6718b8e34f" \t "https://zhuanlan.zhihu.com/p/_blank)

## **布局指南**

在scrollView的滚动布局中，如果使用column组件，并为其添加Expanded扩展子组件的话，这两者会存在冲突。

如果坚持要使用此布局，在column设置mainAxisSize: MainAxisSize.min，同时子组件由Expanded改为Flexible即可。

## **表单、校验**

[https://www.cnblogs.com/pengs...](https://link.zhihu.com/?target=https://www.cnblogs.com/pengshaomin/p/8945720.html" \t "https://zhuanlan.zhihu.com/p/_blank)

## **1、单行文本输入框 TextFormField**

new TextFormField(

maxLength: 32,

onSaved: (val)=> this.\_config = val,

validator: (v)=>(v == null || v.isEmpty)?"请选择配置": null,

decoration: new InputDecoration(

labelText: '配置',

),

),



## **3、多行输入框 keyboardType: TextInputType.multiline,**

keyboardType: TextInputType.multiline,

maxLines: 3,

maxLength: 100,

),



## **4、单选Radio**

new Radio(

groupValue: this.radio,

activeColor: Colors.blue,

value: 'aaa',

onChanged: (String val) {

// val 与 value 的类型对应

this.setState(() {

this.radio = val; // aaa

});

},

),



## **5、复选 CheckBox**

new Checkbox(

value: flutter,

activeColor: Colors.blue,

onChanged: (val) {

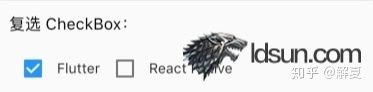
setState(() {

flutter = val;

});

},

),



## **6、switch**

new Switch(

activeColor: Colors.green,

value: flutter,

onChanged: (val) {

setState(() {

flutter = val;

});

},

),



## **7、slider**

new Slider(

value: \_slider,

min: 0.0,

max: 100.0,

onChanged: (val) {

setState(() {

\_slider = val;

});

},

),



## **8、DateTimePicker**

// 设置存储日期的变量

DateTime \_dateTime = new DateTime.now();

// 显示文字Text，设置点击事件，点击后打开日期选择器

new GestureDetector(

onTap: (){

\_showDatePicker();

},

child: new Container(

child: new Text(\_dateTime.toLocal().toString()),

),

),

// 打开日期选择器

void \_showDatePicker() {

\_selectDate(context);

}

Future<Null> \_selectDate(BuildContext context) async {

final DateTime \_picked = await showDatePicker(

context: context,

initialDate: \_dateTime,

firstDate: new DateTime(2016),

lastDate: new DateTime(2050)

);

if(\_picked != null) {

print(\_picked);

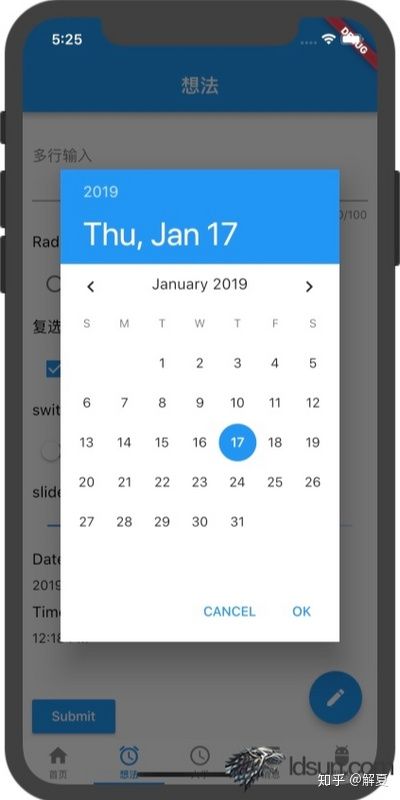
setState(() {

\_dateTime = \_picked;

});

}

}



## **9、TimePIcker**

TimeOfDay \_time = new TimeOfDay.now();

// text显示当前时间

new GestureDetector(

onTap: \_showTimePicker,

child: new Text(\_time.format(context)),

),

// 显示timpicker

void \_showTimePicker(){

\_selectTime(context);

}

Future<Null> \_selectTime(BuildContext context) async {

final TimeOfDay \_picker = await showTimePicker(

context: context,

initialTime: \_time,

);

if(\_picker != null) {

print(\_picker);

setState(() {

\_time = \_picker;

});

}

}



Toast/showSnackBar

## **showSnackBar:**

[https://material.io/design/co...](https://link.zhihu.com/?target=https://material.io/design/components/snackbars.html%23usage" \t "https://zhuanlan.zhihu.com/p/_blank)

void \_showToast(BuildContext context) {

final scaffold = Scaffold.of(context);

scaffold.showSnackBar(

SnackBar(

content: const Text('Added to favorite'),

action: SnackBarAction(

label: 'UNDO',

onPressed: scaffold.hideCurrentSnackBar

),

),

);

}

## **Toast:**

[https://github.com/PonnamKart...](https://link.zhihu.com/?target=https://github.com/PonnamKarthik/FlutterToast" \t "https://zhuanlan.zhihu.com/p/_blank)

void \_showToast(String title) {

Fluttertoast.showToast(

msg: title,

toastLength: Toast.LENGTH\_SHORT,

gravity: ToastGravity.CENTER,

timeInSecForIos: 1,

backgroundColor: Color.fromRGBO(0, 0, 0, 0.85),

textColor: Colors.white

);

}

## **Popover/popup**

## **popup: CupertinoActionSheet组件 -- Actionsheet in flutter**

[http://flatteredwithflutter.c...](https://link.zhihu.com/?target=http://flatteredwithflutter.com/actionsheet-in-flutter/" \t "https://zhuanlan.zhihu.com/p/_blank)

import 'package:flutter/cupertino.dart';

new MaterialButton(

onPressed: () {

\_showActionSheet();

},

child: new Text('show ActionSheet', style: TextStyle(color: Colors.white),),

color: Colors.greenAccent,

),

void \_showActionSheet() {

showCupertinoModalPopup(

context: context,

builder: (BuildContext context) => actionSheet(),

).then((value) {

Scaffold.of(context).showSnackBar(new SnackBar(

content: new Text('You clicked $value'),

));

});

}

Widget actionSheet(){

return new CupertinoActionSheet(

title: new Text('title'),

message: const Text('your options are'),

actions: <Widget>[

CupertinoActionSheetAction(

child: const Text('yes'),

onPressed: (){

Navigator.pop(context, 'yes');

},

),

CupertinoActionSheetAction(

child: const Text('no'),

onPressed: (){

Navigator.pop(context, 'no');

},

)

],

cancelButton: CupertinoActionSheetAction(

child: new Text('cancel'),

onPressed: () {

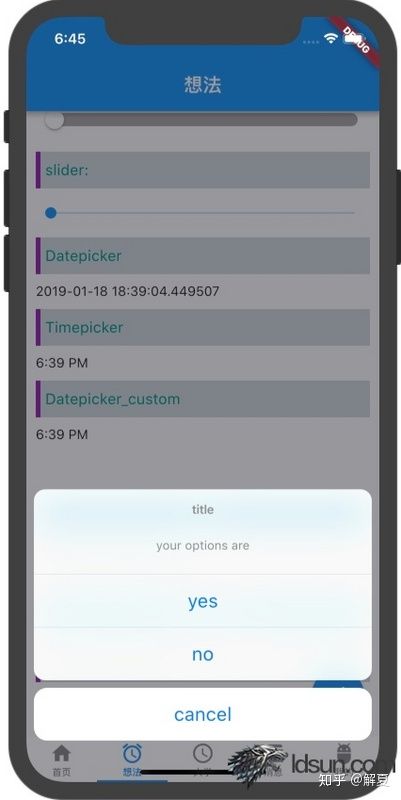
Navigator.pop(context, 'Cancel');

},

),

);

}



## **Dismissible 滑动删除**

[https://flutter.io/docs/cookb...](https://link.zhihu.com/?target=https://flutter.io/docs/cookbook/gestures/dismissible" \t "https://zhuanlan.zhihu.com/p/_blank)

new Dismissible(

// Each Dismissible must contain a Key. Keys allow Flutter to

// uniquely identify Widgets.

key: Key(item),

onDismissed: (direction) {

setState(() {

items.removeAt(index);

});

// Then show a snackbar!

Scaffold.of(context)

.showSnackBar(SnackBar(content: Text("$item dismissed")));

},

// Show a red background as the item is swiped away

background: Container(color: Colors.red),

child: ListTile(title: Text('$item')),

);

## **Swipe 左滑右滑删除**

Widget \_swipe(int i, String title, String desc) {

return new Slidable(

delegate: new SlidableDrawerDelegate(),

actionExtentRatio: 0.25,

child: new Container(

color: Colors.white,

child: new GestureDetector(

onTap: (){},

onDoubleTap: (){},

onLongPress: (){},

child: new ListTile(

leading: new CircleAvatar(

backgroundColor: Colors.grey[200],

child: new Text(

'$i',

style: TextStyle(color: Colors.orange),

),

foregroundColor: Colors.white,

),

title: new Text(

'$title',

maxLines: 1,

overflow: TextOverflow.ellipsis,

style: TextStyle(color: Colors.black87, fontSize: 16),

),

subtitle: new Text(

'$desc',

style: TextStyle(color: Colors.blue[300]),

),

),

)

),

actions: <Widget>[

new IconSlideAction(

caption: 'Archive',

color: Colors.blue,

icon: Icons.archive,

onTap: () => \_showSnackBar('Archive'),

),

new IconSlideAction(

caption: 'Share',

color: Colors.indigo,

icon: Icons.share,

onTap: () => \_showSnackBar('Share'),

),

],

secondaryActions: <Widget>[

new IconSlideAction(

caption: 'More',

color: Colors.black45,

icon: Icons.more\_horiz,

onTap: () => \_showSnackBar('More'),

),

new IconSlideAction(

caption: 'Delete',

color: Colors.red,

icon: Icons.delete,

onTap: () => \_showSnackBar('Delete'),

),

],

);

}



## **常用手势 GestureDetector**

new GestureDetector(

onTap: (){\_showToast('点击: $i');},

onDoubleTap: (){\_showToast('连点: $i');},

onLongPress: (){\_showToast('长按: $i');},

)

## **image\_picker： (最常用场景，从相册选择或手机拍照得到照片)**

dynamic \_picture;

dynamic \_gallery;

new FlatButton.icon(

icon: Icon(Icons.camera),

label: Text('选择头像'),

onPressed: (){

\_optionsDialogBox();

},

),

Future<void> \_optionsDialogBox() {

return showDialog(context: context,

builder: (BuildContext context) {

return AlertDialog(

content: new SingleChildScrollView(

child: new ListBody(

children: <Widget>[

GestureDetector(

child: new Text('Take a picture'),

onTap: openCamera,

),

Padding(

padding: EdgeInsets.all(8.0),

),

GestureDetector(

child: new Text('Select from gallery'),

onTap: openGallery,

),

],

),

),

);

});

}

void openCamera() async {

Navigator.of(context).pop();

var picture = await ImagePicker.pickImage(

source: ImageSource.camera,

);

setState(() {

\_picture = picture;

});

}

void openGallery() async {

Navigator.of(context).pop();

var gallery = await ImagePicker.pickImage(

source: ImageSource.gallery,

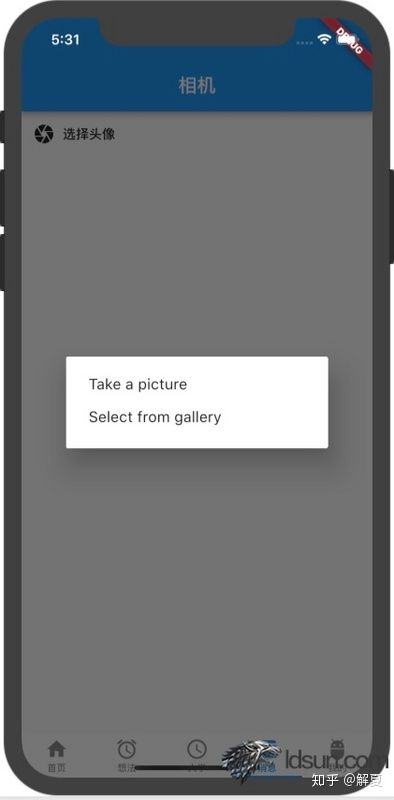
);

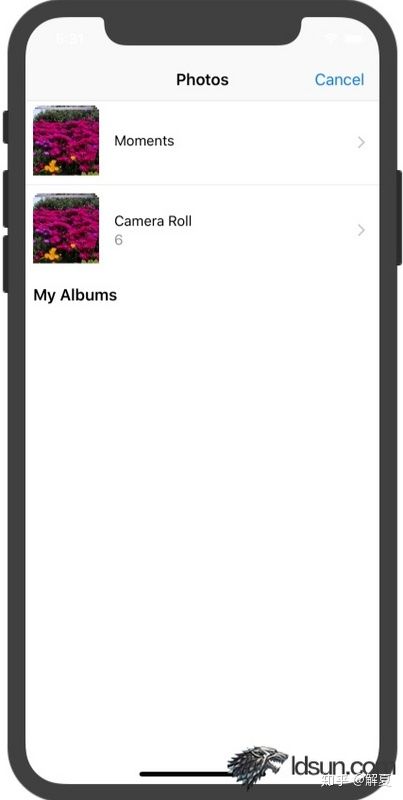
setState(() {

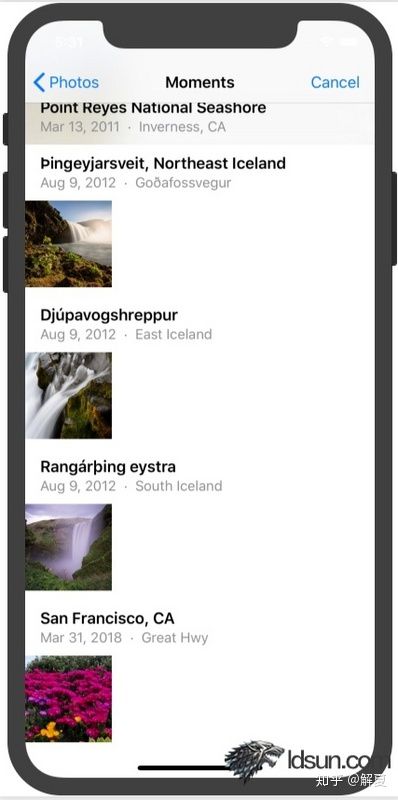
\_gallery = gallery;

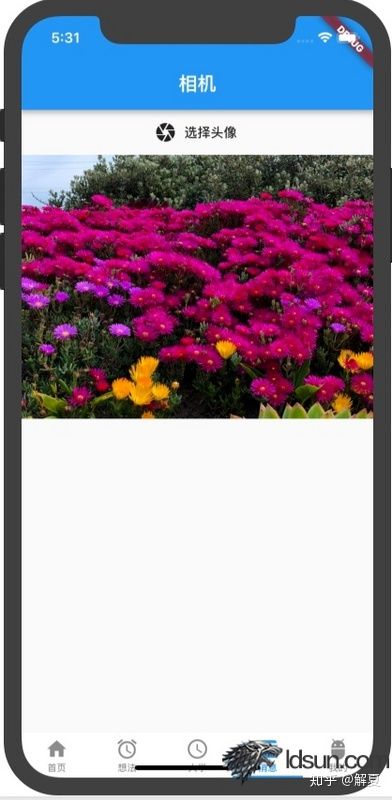
});

}









camera： (高阶用法，打开相机，实时获取相机流，可以定制拍照、录像等按钮。可用于相机扫码、实时识别、直播等场景)

[https://pub.dartlang.org/pack...](https://link.zhihu.com/?target=https://pub.dartlang.org/packages/camera" \t "https://zhuanlan.zhihu.com/p/_blank)

camera: ^0.2.9

import 'package:camera/camera.dart';

class \_CameraState extends State<CameraWidget> {

List<CameraDescription> cameras;

CameraController controller;

bool \_isReady = false;

@override

void initState() {

super.initState();

\_setupCameras();

}

Future<void> \_setupCameras() async {

try {

// initialize cameras.

cameras = await availableCameras();

// initialize camera controllers.

controller = new CameraController(cameras[0], ResolutionPreset.medium);

await controller.initialize();

} on CameraException catch (\_) {

// do something on error.

}

if (!isMounted) return;

setState(() {

\_isReady = true;

});

}

Widget build(BuildContext context) {

if (!\_isReady) return new Container();

return new Container(

height: 200,

child: AspectRatio(

aspectRatio: controller.value.aspectRatio,

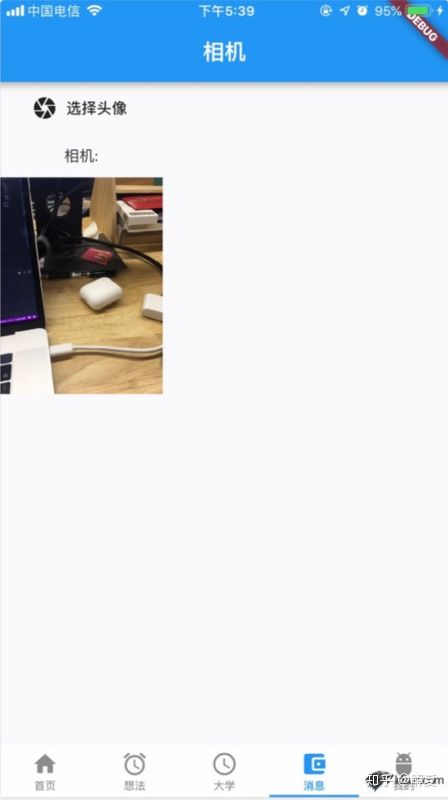
child: CameraPreview(controller),

),

)

}

}



## **video player**

video\_player: ^0.8.0

import 'package:video\_player/video\_player.dart';

VideoPlayerController \_controller;

bool \_isPlaying = false;

@override

void initState() {

super.initState();

\_controller = VideoPlayerController.network(

'https://www.quirksmode.org/html5/videos/big\_buck\_bunny.mp4',

)

..addListener(() {

final bool isPlaying = \_controller.value.isPlaying;

if (isPlaying != \_isPlaying) {

setState(() {

\_isPlaying = isPlaying;

});

}

})

..initialize().then((\_) {

// Ensure the first frame is shown after the video is initialized, even before the play button has been pressed.

setState(() {});

});

}

@override

void dispose() {

\_controller?.dispose();

super.dispose();

}

// 显示、控制

\_controller.value.initialized

? AspectRatio(

aspectRatio: \_controller.value.aspectRatio,

child: new Container(

padding: EdgeInsets.all(10),

color: Colors.black,

child: VideoPlayer(\_controller),

),

)

: Container(

child: new Text('视频加载中~'),

),

new FlatButton.icon(

label: Text('播放/暂停'),

icon: Icon(

\_controller.value.isPlaying ? Icons.pause : Icons.play\_arrow,

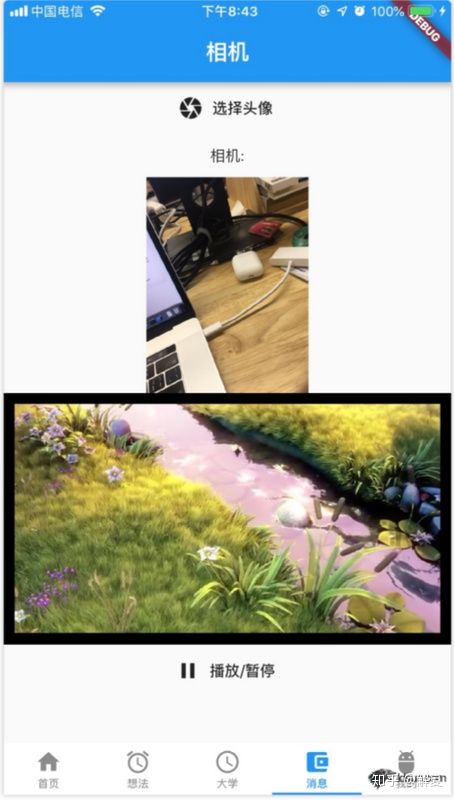
),

onPressed: \_controller.value.isPlaying

? \_controller.pause

: \_controller.play,

)



## **AudioPlayer**

