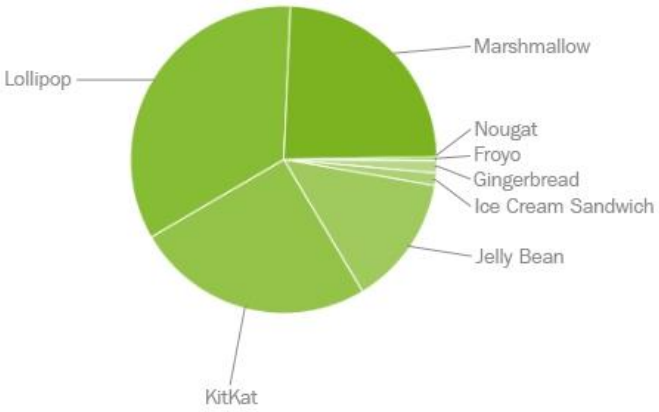
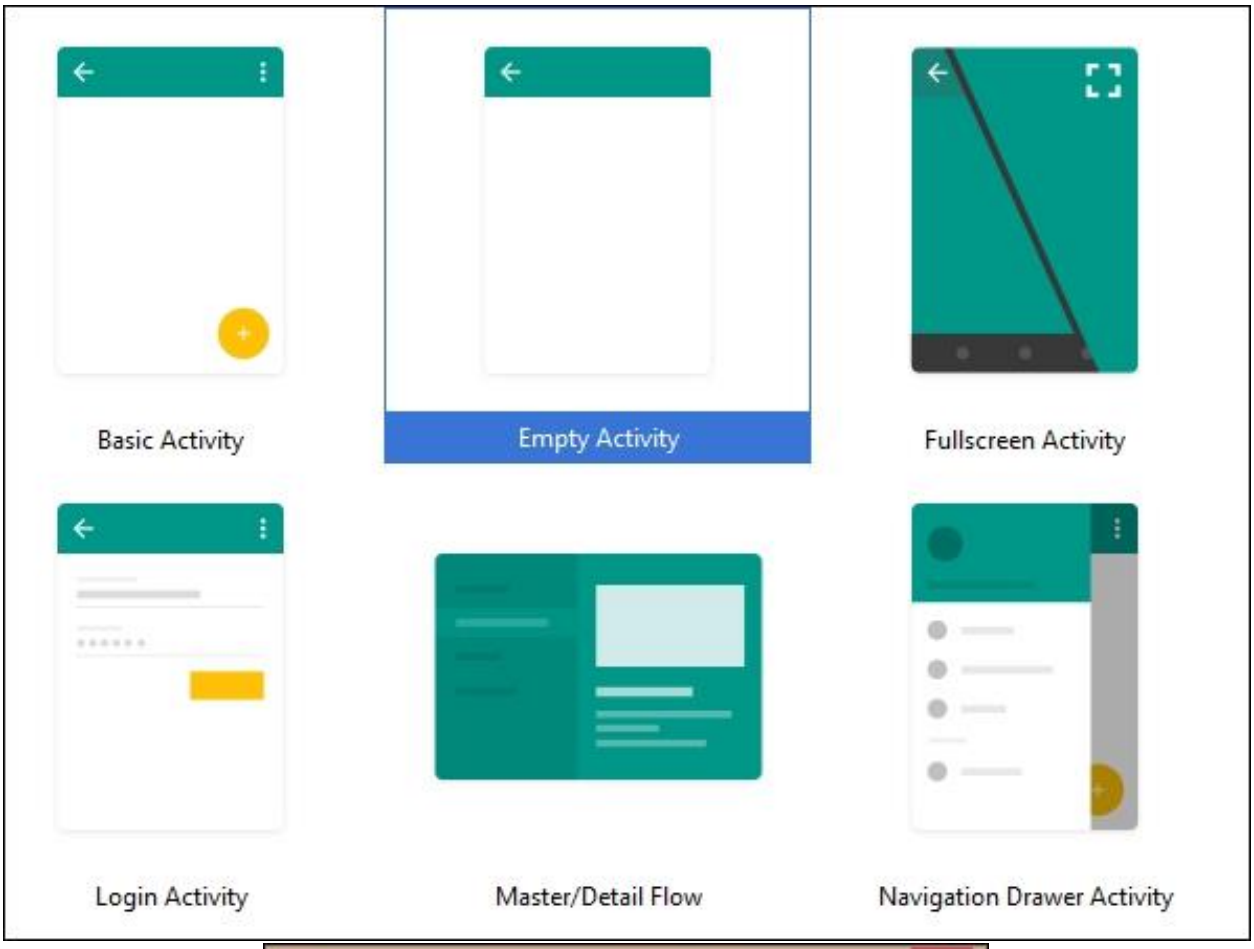
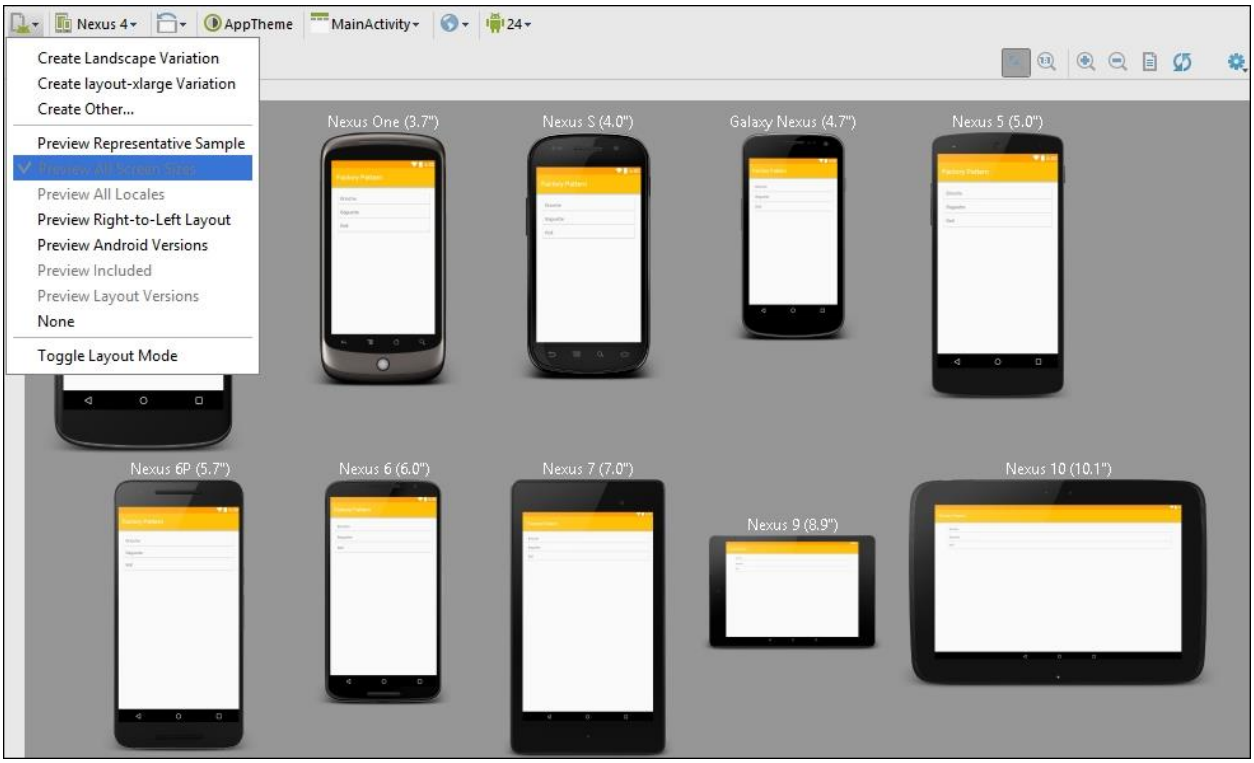
# Chapter 1: Design Patterns

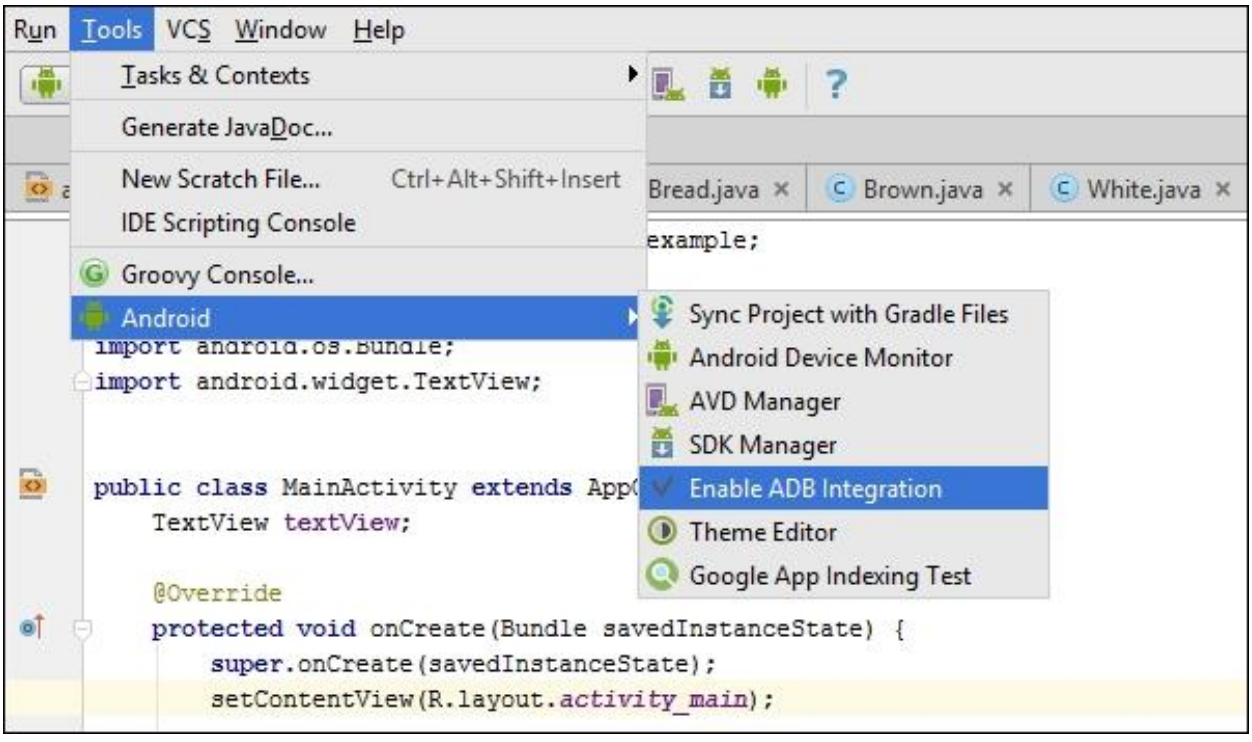
| Version | Codename | API | Distribution |
|---|---|---|---|
| 2.2 | Froyo | 8 | 0.1% |
| 2.3.3 - 2.3.7 | Gingerbread | 10 | 1.3% |
| 4.0.3 - 4.0.4 | Ice Cream Sandwich | 15 | 1.3% |
| 4.1.x | Jelly Bean | 16 | 4.9% |
| 4.2.x | | 17 | 6.8% |
| 4.3 | | 18 | 2.0% |
| 4.4 | KitKat | 19 | 25.2% |
| 5.0 | Lollipop | 21 | 11.3% |
| 5.1 | | 22 | 22.8% |
| 6.0 | Marshmallow | 23 | 24.0% |
| 7.0 | Nougat | 24 | 0.3% |

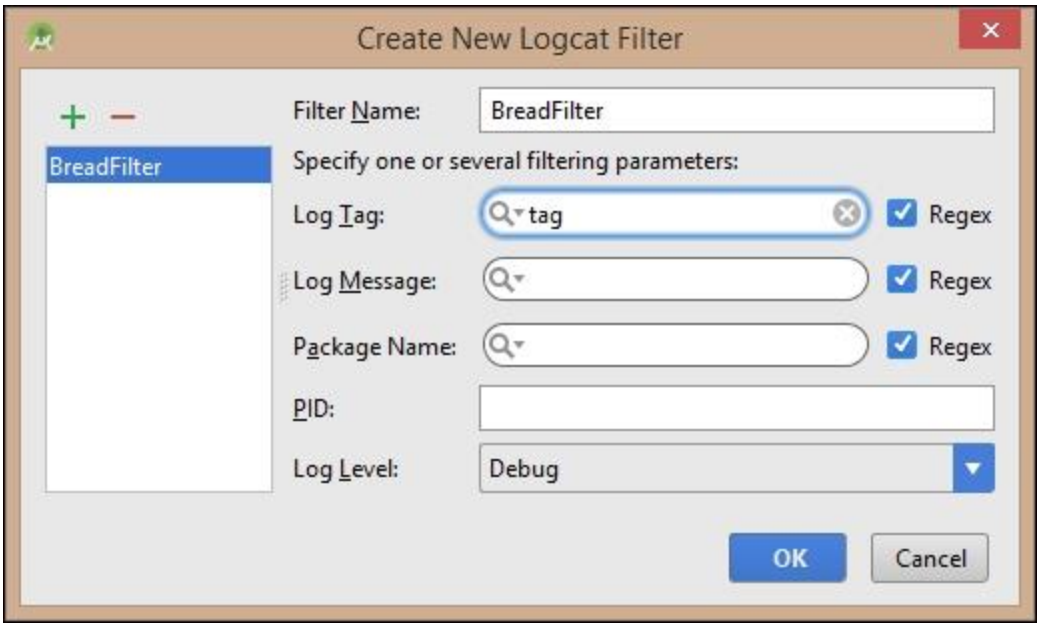


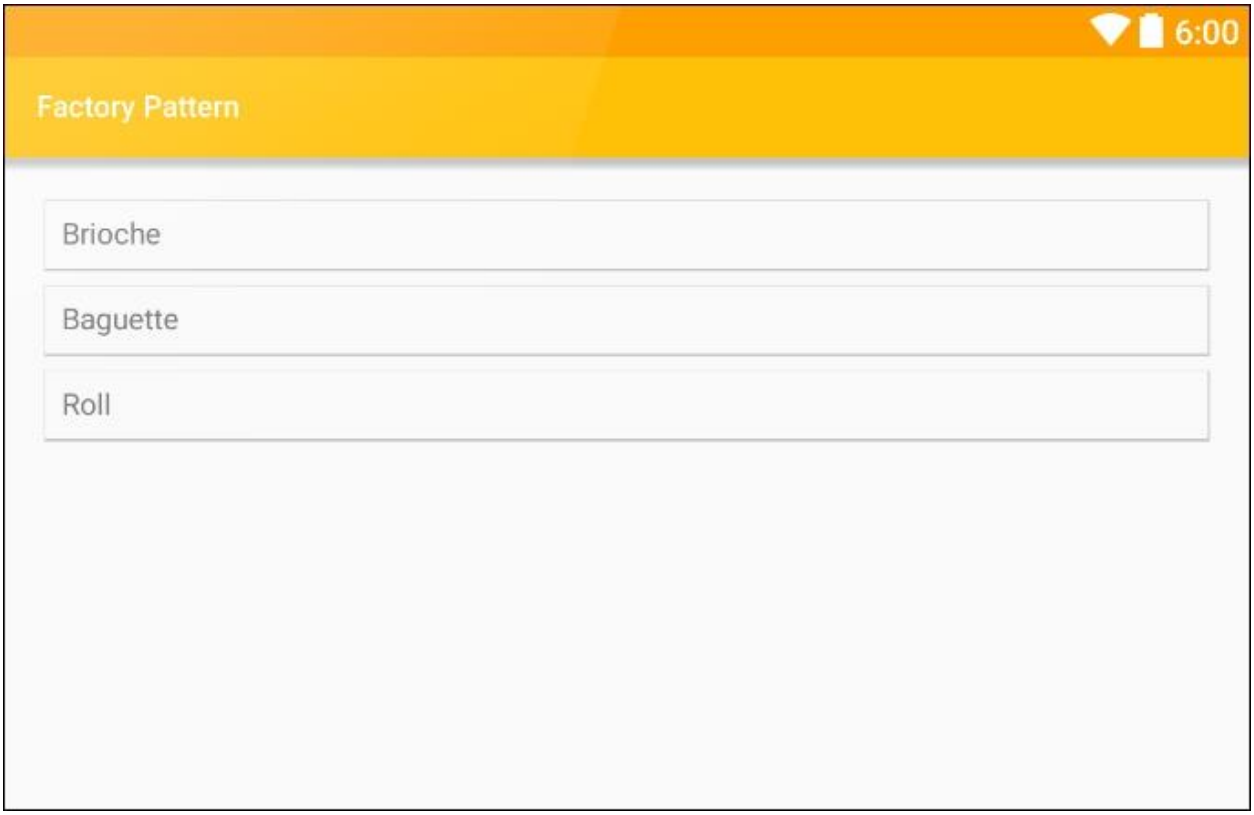

Basic Activity
Empty Activity
Fullscreen Activity
Login Activity
Master/Detail Flow
Navigation Drawer Activity

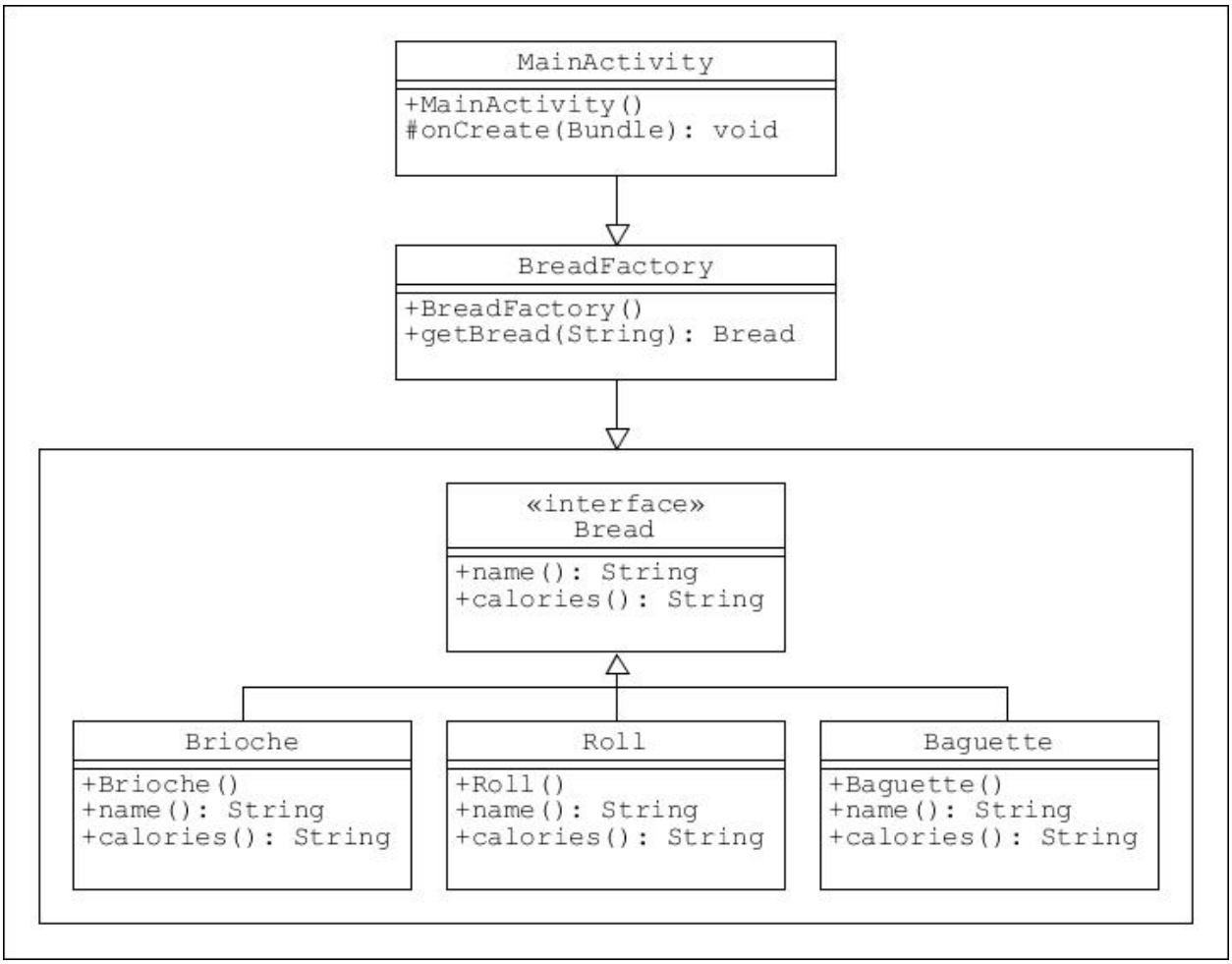
Create New Class
Name:
Bread
Kind:
Interface
OK
Cancel

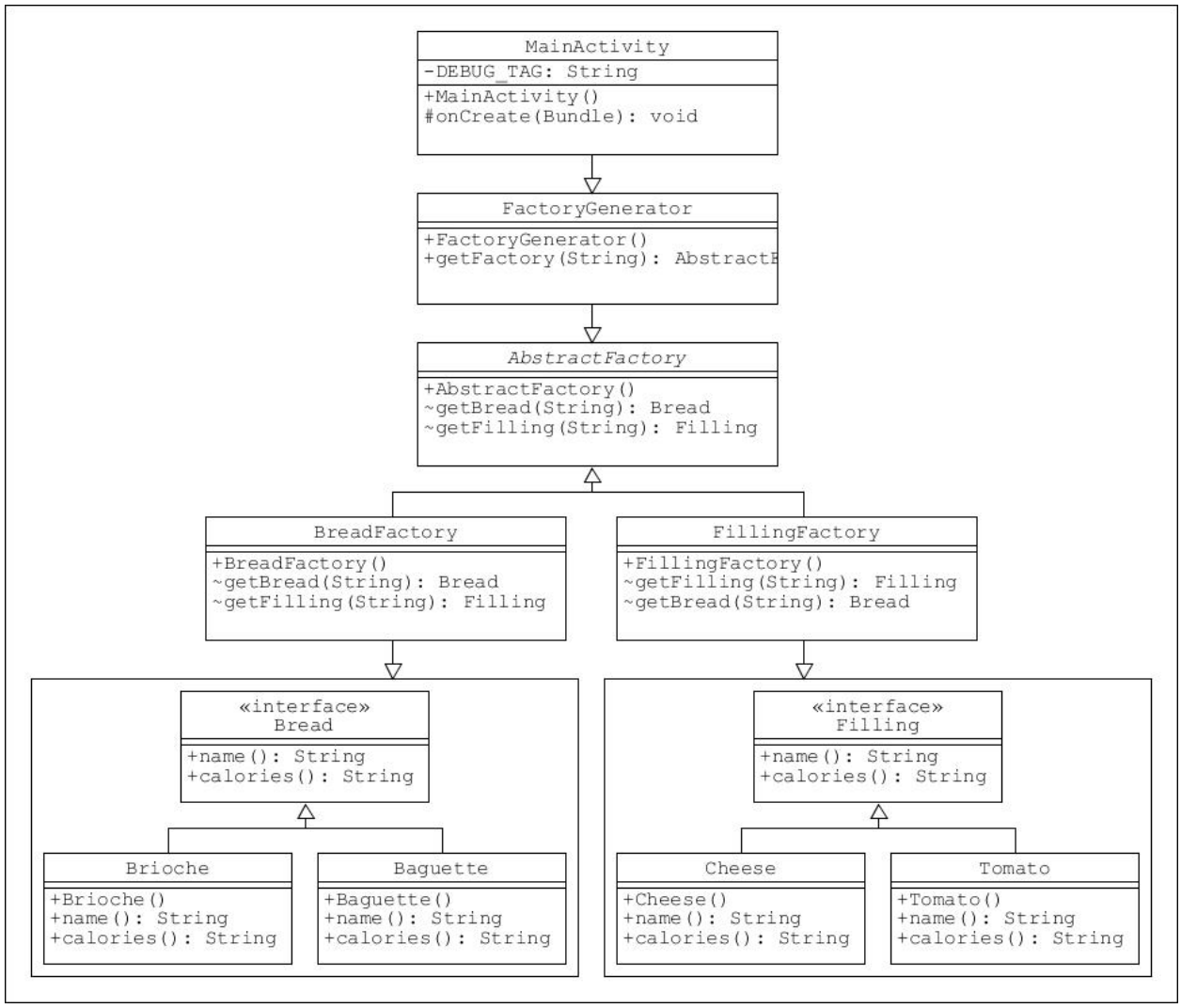

Nexus 4
AppTheme
MainActivity
24
Create Landscape Variation
Create layout-xlarge Variation
Create Other...
Preview Representative Sample
Preview All Screen Sizes
Preview All Locales
Preview Right-to-Left Layout
Preview Android Versions
Preview Included
Preview Layout Versions
None
Toggle Layout Mode
Nexus One (3.7")
Nexus S (4.0")
Galaxy Nexus (4.7")
Nexus 5 (5.0")
Nexus 6P (5.7")
Nexus 6 (6.0")
Nexus 7 (7.0")
Nexus 9 (8.9")
Nexus 10 (10.1")


Run
Tools
VCS
Window
Help
Tasks & Contexts
Generate JavaDoc...
New Scratch File... Ctrl+Alt+Shift+Insert
IDE Scripting Console
Groovy Console...
Android
Sync Project with Gradle Files
Android Device Monitor
AVD Manager
SDK Manager
Enable ADB Integration
Theme Editor
Google App Indexing Test
Bread.java
Brown.java
White.java
example;
import android.os.Bundle;
import android.widget.TextView;
public class MainActivity extends App
TextView textView;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);

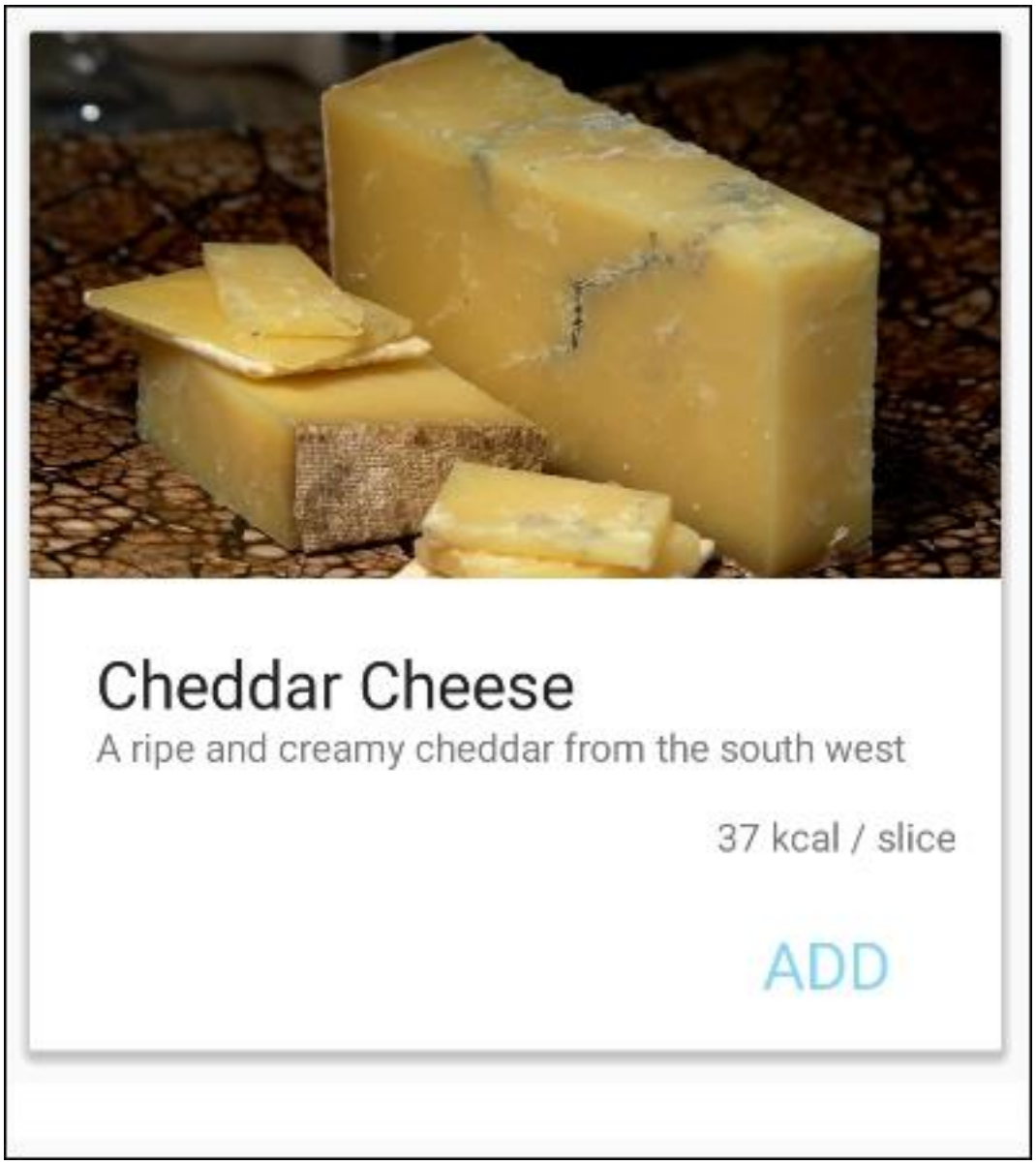

Create New Logcat Filter
Filter Name:
BreadFilter
BreadFilter
Specify one or several filtering parameters:
Log Tag:
tag
Regex
Log Message:
Regex
Package Name:
Regex
PID:
Log Level:
Debug
OK
Cancel

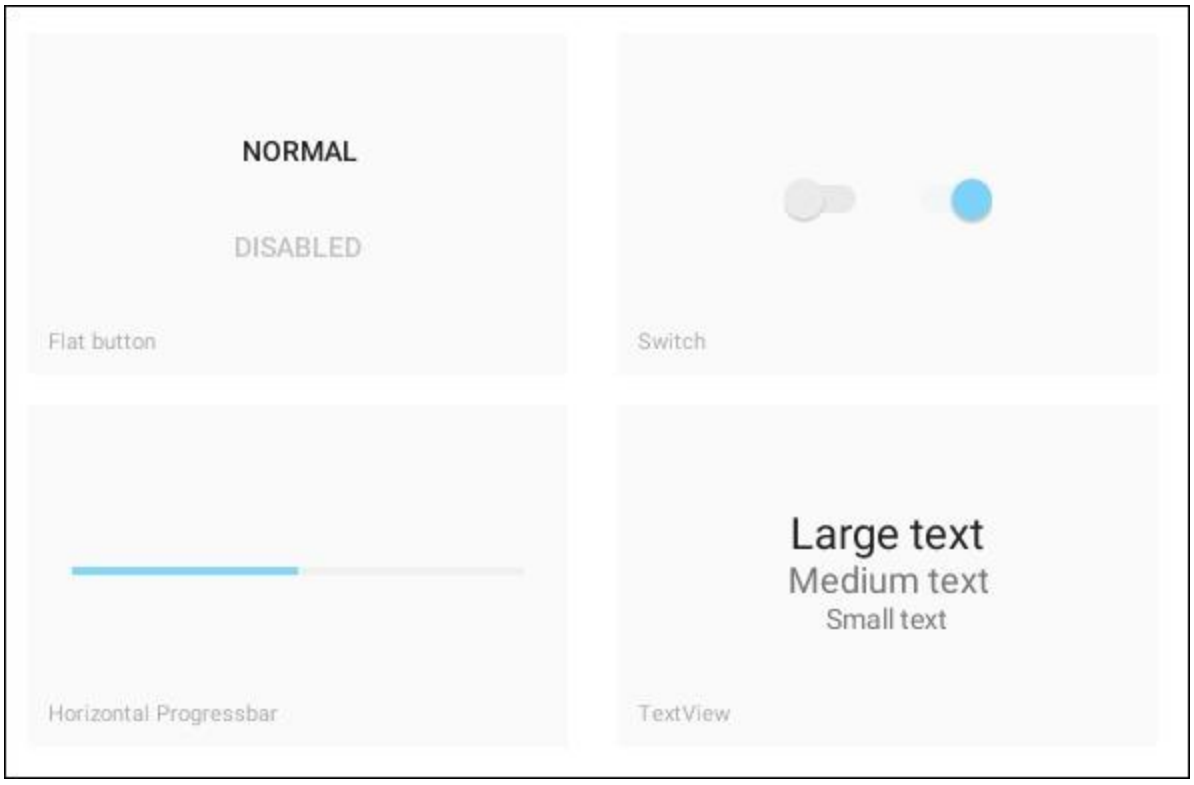

6:00
Factory Pattern
Brioche
Baguette
Roll


MainActivity
+MainActivity()
#onCreate(Bundle): void
BreadFactory
+BreadFactory()
+getBread(String): Bread
«interface»
Bread
+name(): String
+calories(): String
Brioche
+Brioche()
+name(): String
+calories(): String
Roll
+Roll()
+name(): String
+calories(): String
Baguette
+Baguette()
+name(): String
+calories(): String


MainActivity
-DEBUG_TAG: String
+MainActivity()
#onCreate(Bundle): void
FactoryGenerator
+FactoryGenerator()
+getFactory(String): AbstractF
AbstractFactory
+AbstractFactory()
~getBread(String): Bread
~getFilling(String): Filling
BreadFactory
+BreadFactory()
~getBread(String): Bread
~getFilling(String): Filling
FillingFactory
+FillingFactory()
~getFilling(String): Filling
~getBread(String): Bread
«interface»
Bread
+name(): String
+calories(): String
«interface»
Filling
+name(): String
+calories(): String
Brioche
+Brioche()
+name(): String
+calories(): String
Baguette
+Baguette()
+name(): String
+calories(): String
Cheese
+Cheese()
+name(): String
+calories(): String
Tomato
+Tomato()
+name(): String
+calories(): String

# Chapter 2: Creational Patterns






| Yellow | |
|---|---|
| 500 | #FFEB3B |
| 50 | #FFFDE7 |
| 100 | #FFF9C4 |
| 200 | #FFF59D |
| 300 | #FFF176 |
| 400 | #FFEE58 |
| 500 | #FFEB3B |
| 600 | #FDD835 |
| 700 | #FBC02D |
| 800 | #F9A825 |
| 900 | #F57F17 |

| Amber | |
|---|---|
| 500 | #FFC107 |
| 50 | #FFF8E1 |
| 100 | #FFECB3 |
| 200 | #FFE082 |
| 300 | #FFD54F |
| 400 | #FFCA28 |
| 500 | #FFC107 |
| 600 | #FFB300 |
| 700 | #FFA000 |
| 800 | #FF8F00 |
| 900 | #FF6F00 |

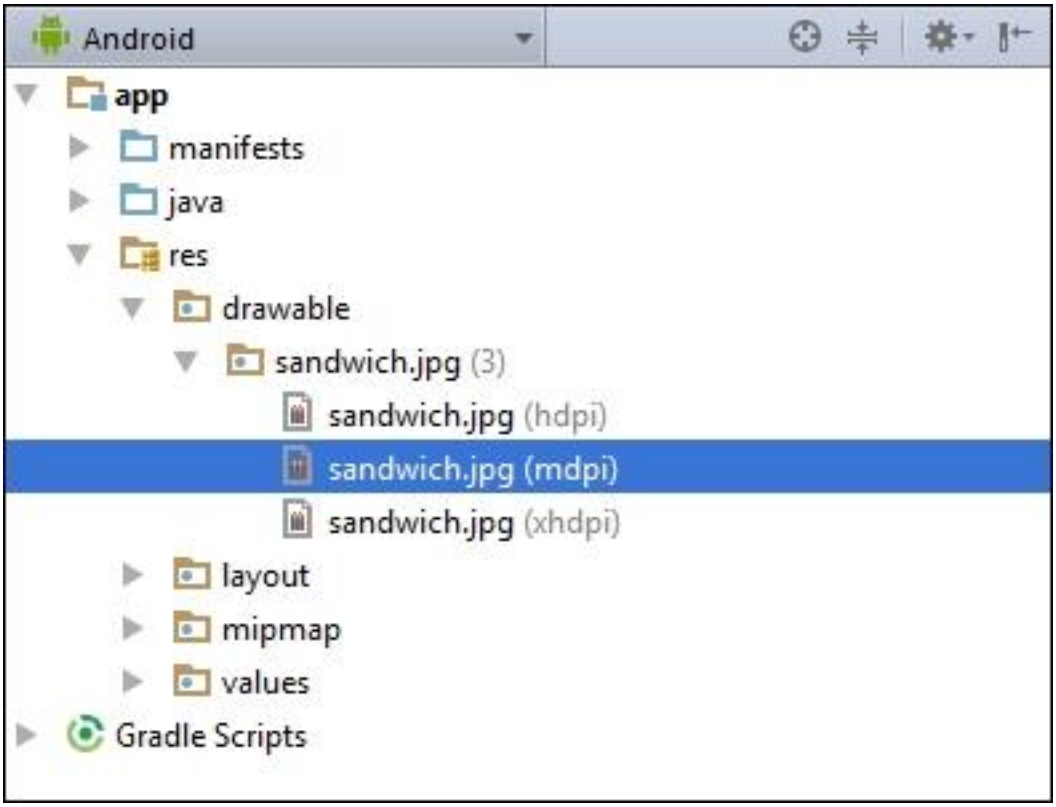
| Orange | |
|---|---|
| 500 | #FF9800 |
| 50 | #FFF3E0 |
| 100 | #FFE0B2 |
| 200 | #FFCC80 |
| 300 | #FFB74D |
| 400 | #FFA726 |
| 500 | #FF9800 |
| 600 | #FB8C00 |
| 700 | #F57C00 |
| 800 | #EF6C00 |
| 900 | #E65100 |

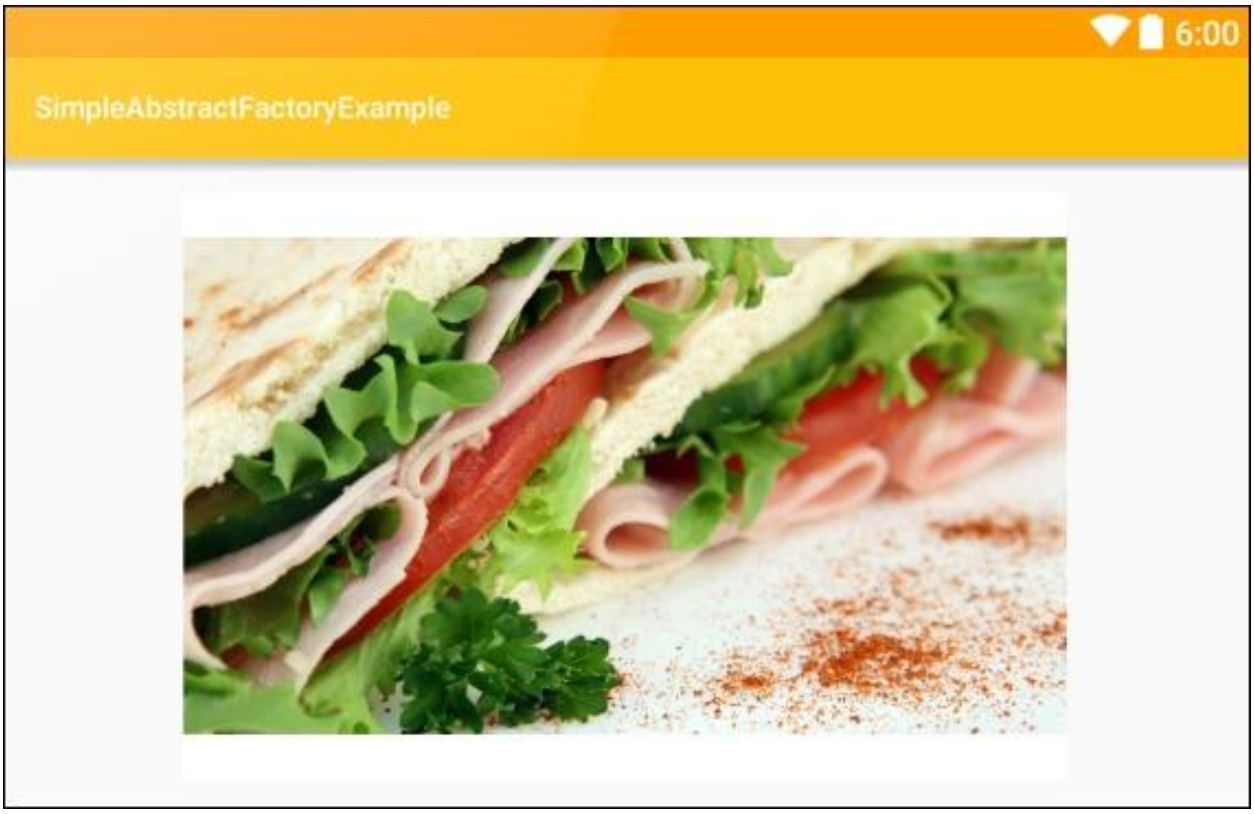
**Primary text** on a *dark background* is 87% opaque black: `#DE000000`

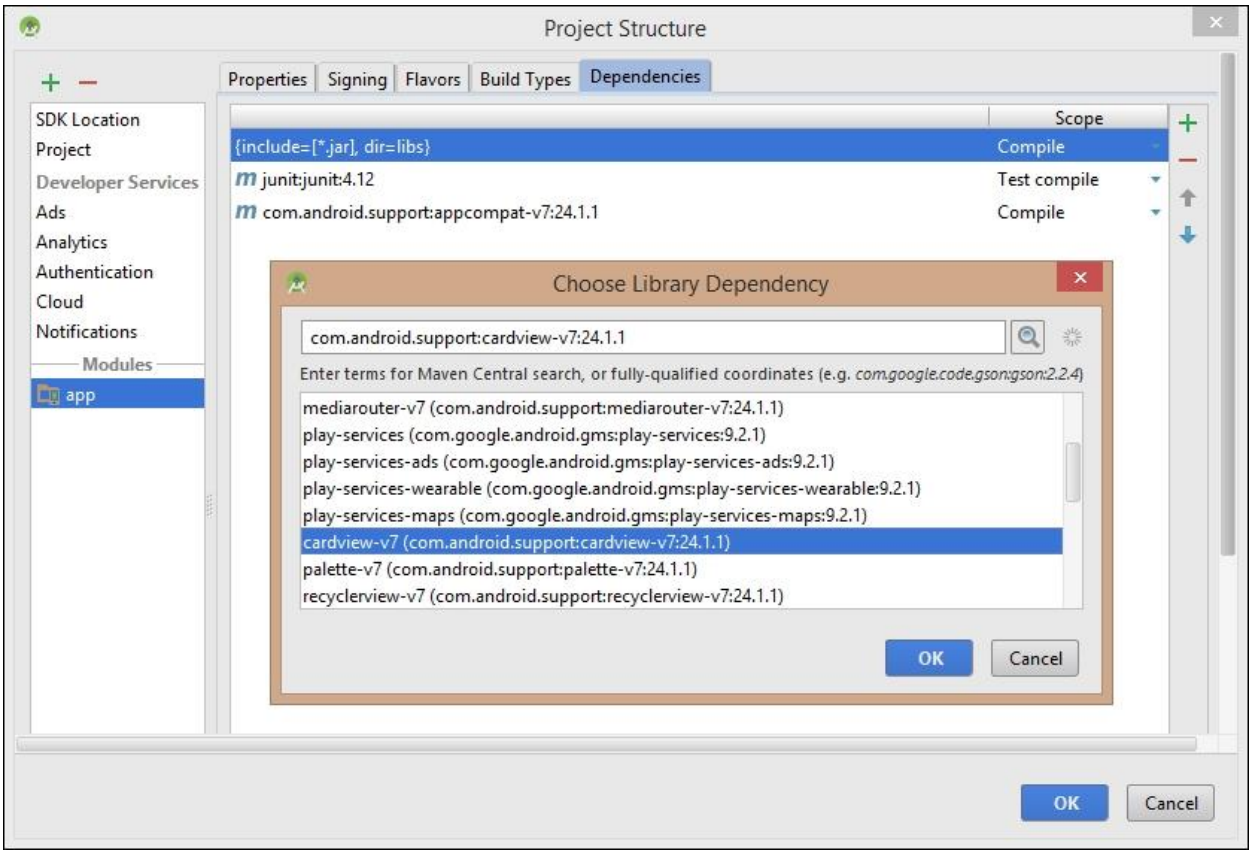
**Secondary text** on a *dark background* is 54% opaque black: `#8A000000`

**Primary text** on a *light background* is 100% opaque white: `#FFFFFFFF`

**Secondary text** on a *light background* is 70% opaque white: `#B3FFFFFF`




6:00
SimpleAbstractFactoryExample


Project Structure
Properties
Signing
Flavors
Build Types
Dependencies
SDK Location
Project
Developer Services
Ads
Analytics
Authentication
Cloud
Notifications
Modules
app
Scope
{include=[*.jar], dir=libs}
Compile
junit:junit:4.12
Test compile
com.android.support:appcompat-v7:24.1.1
Compile
Choose Library Dependency
com.android.support:cardview-v7:24.1.1
Enter terms for Maven Central search, or fully-qualified coordinates (e.g. com.google.code.gson:gson:2.2.4)
mediarouter-v7 (com.android.support:mediarouter-v7:24.1.1)
play-services (com.google.android.gms:play-services:9.2.1)
play-services-ads (com.google.android.gms:play-services-ads:9.2.1)
play-services-wearable (com.google.android.gms:play-services-wearable:9.2.1)
play-services-maps (com.google.android.gms:play-services-maps:9.2.1)
cardview-v7 (com.android.support:cardview-v7:24.1.1)
palette-v7 (com.android.support:palette-v7:24.1.1)
recyclerview-v7 (com.android.support:recyclerview-v7:24.1.1)
OK
Cancel
OK
Cancel

Component Tree

- Device Screen
  - **card_content** (LinearLayout) (vertical)
    - **image_view** (ImageView) - @drawable/cheddar
    - **text_title** (TextView) - @string/filling
    - **text_description** (TextView) - @string/filling_description
    - **text_calories** (TextView) - @string/calories
    - **text_add** (TextView) - @string/action

## Cheddar Cheese

A ripe and creamy cheddar from the south west

37 kcal / slice

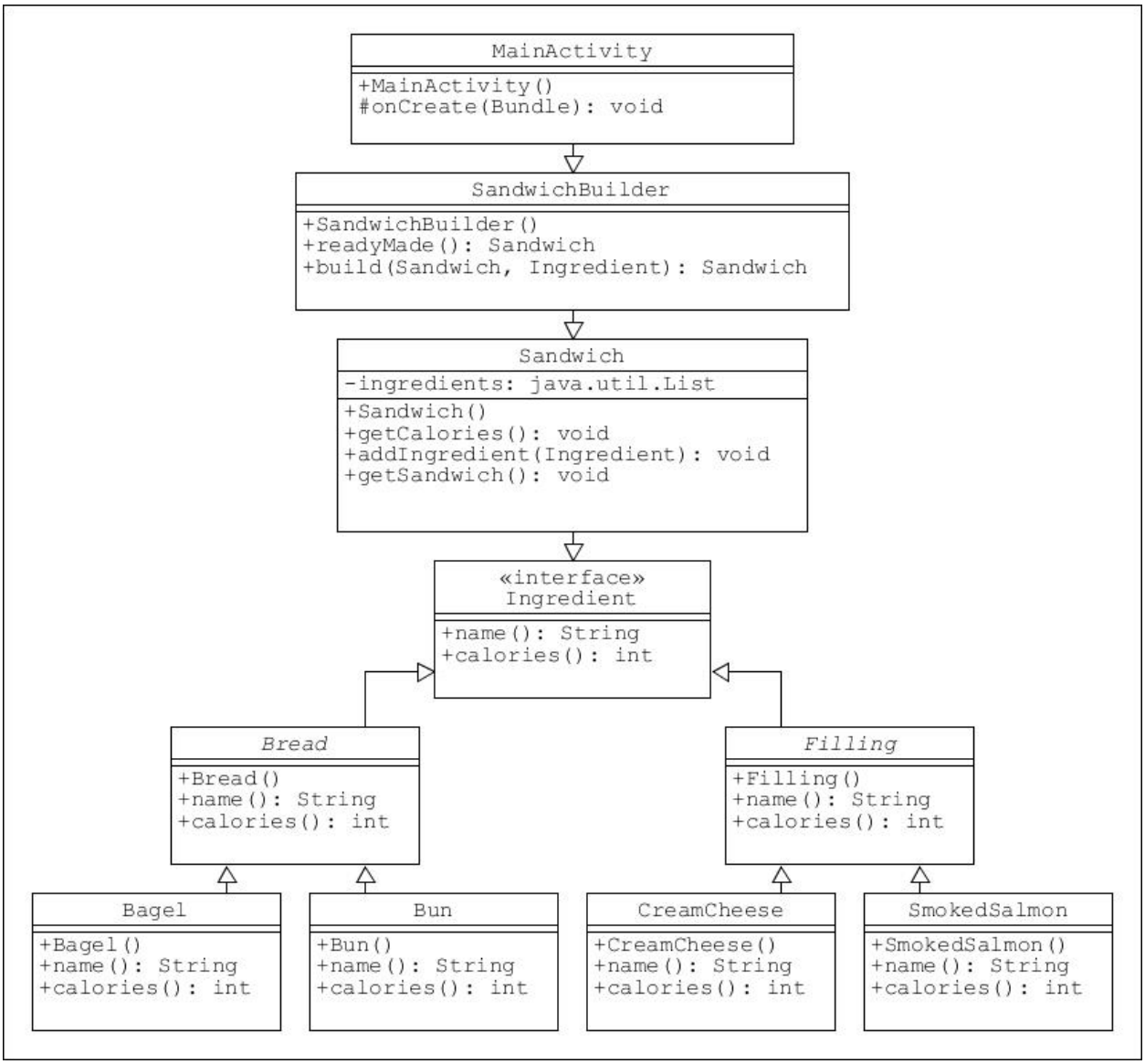
ADD

## Cheddar Cheese

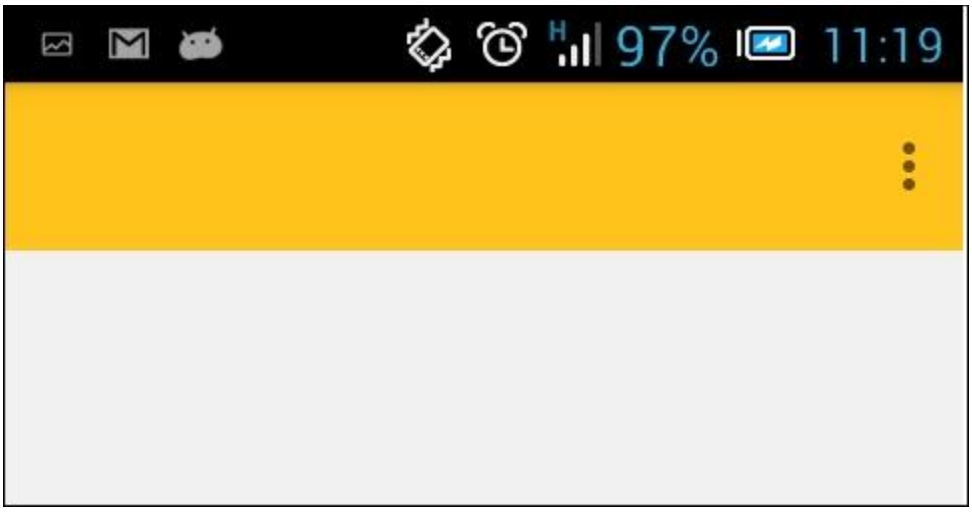
A ripe and creamy cheddar from the south west

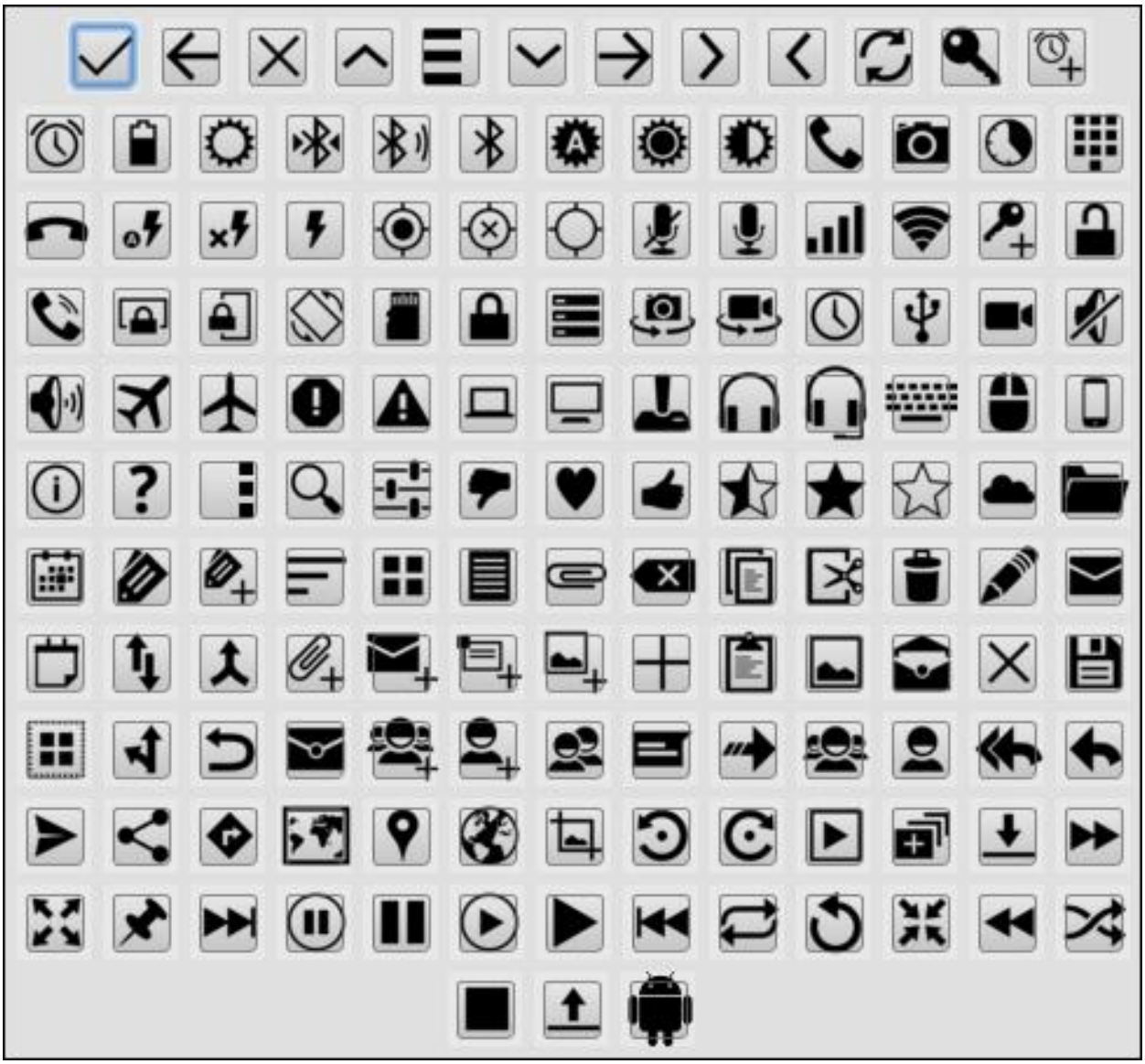
37 kcal / slice

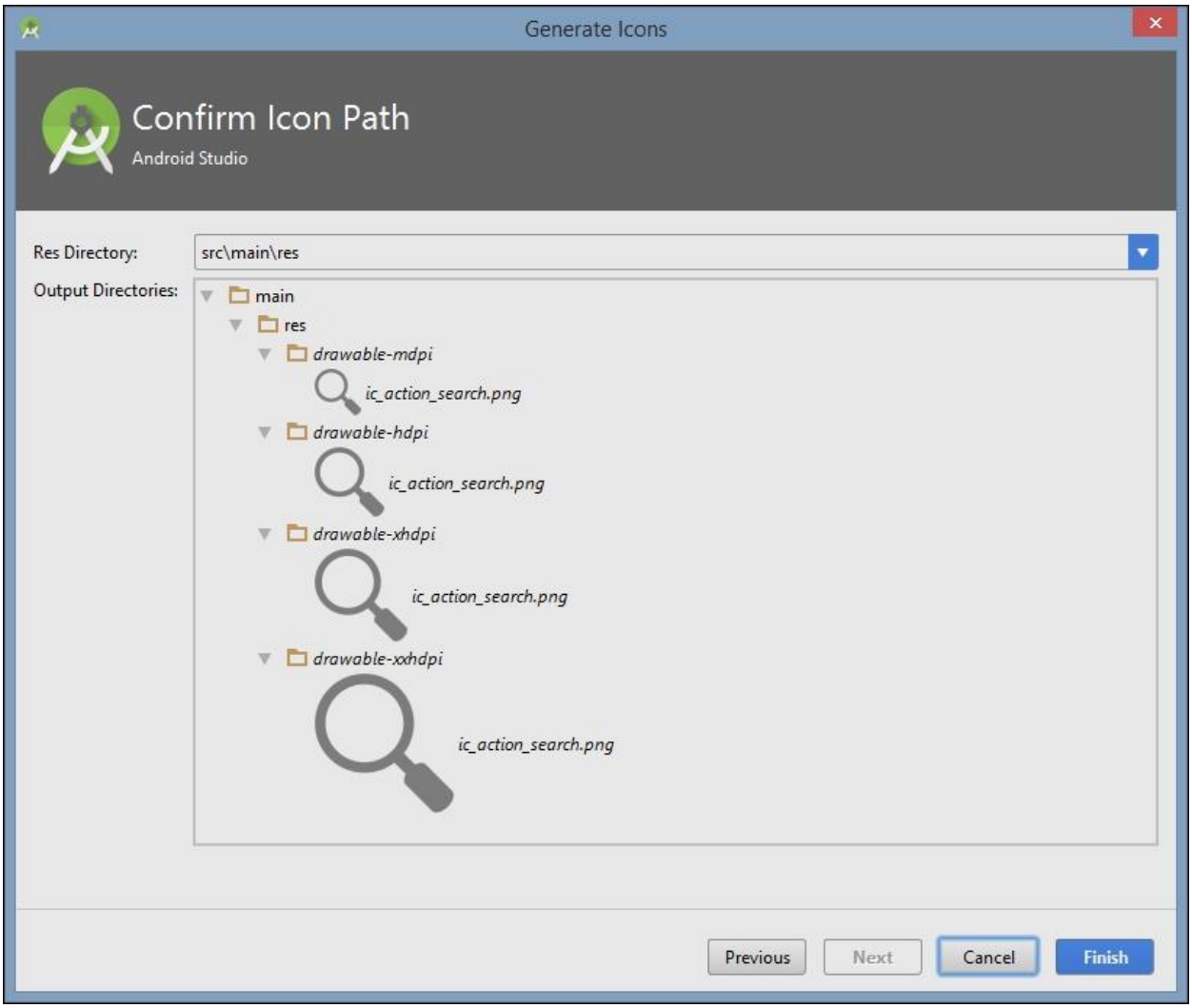
ADD

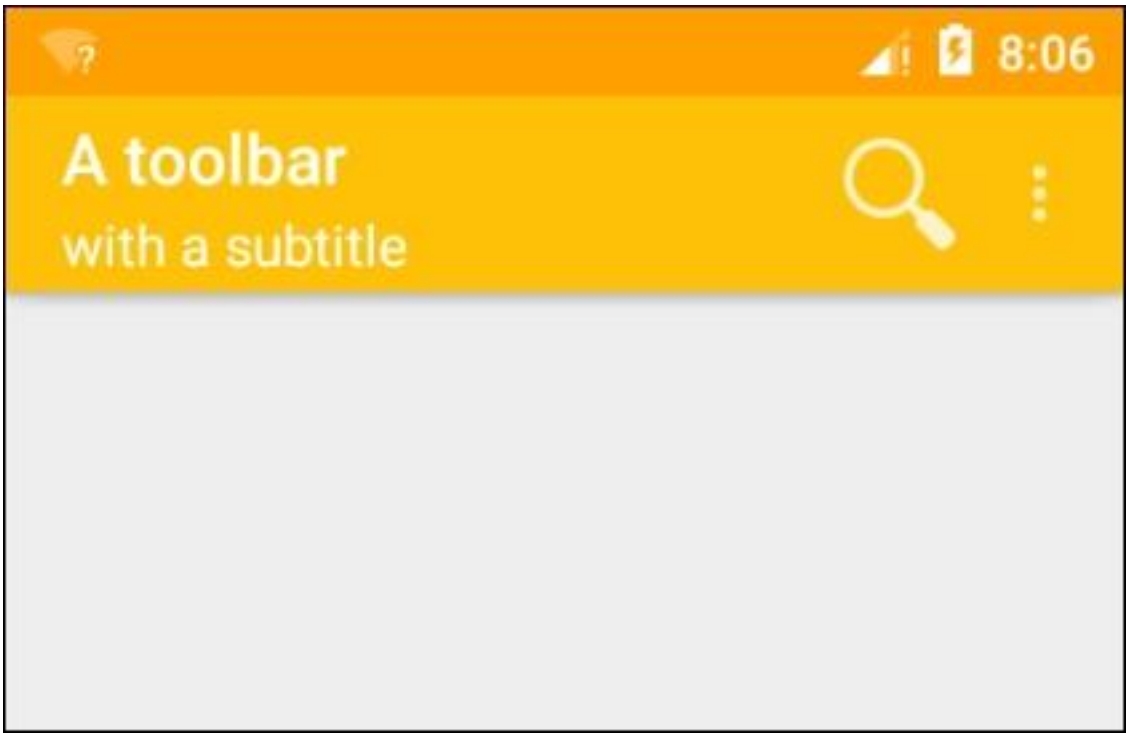

MainActivity
+MainActivity()
#onCreate(Bundle): void
SandwichBuilder
+SandwichBuilder()
+readyMade(): Sandwich
+build(Sandwich, Ingredient): Sandwich
Sandwich
-ingredients: java.util.List
+Sandwich()
+getCalories(): void
+addIngredient(Ingredient): void
+getSandwich(): void
«interface»
Ingredient
+name(): String
+calories(): int
Bread
+Bread()
+name(): String
+calories(): int
Filling
+Filling()
+name(): String
+calories(): int
Bagel
+Bagel()
+name(): String
+calories(): int
Bun
+Bun()
+name(): String
+calories(): int
CreamCheese
+CreamCheese()
+name(): String
+calories(): int
SmokedSalmon
+SmokedSalmon()
+name(): String
+calories(): int

# Chapter 3: Material Patterns

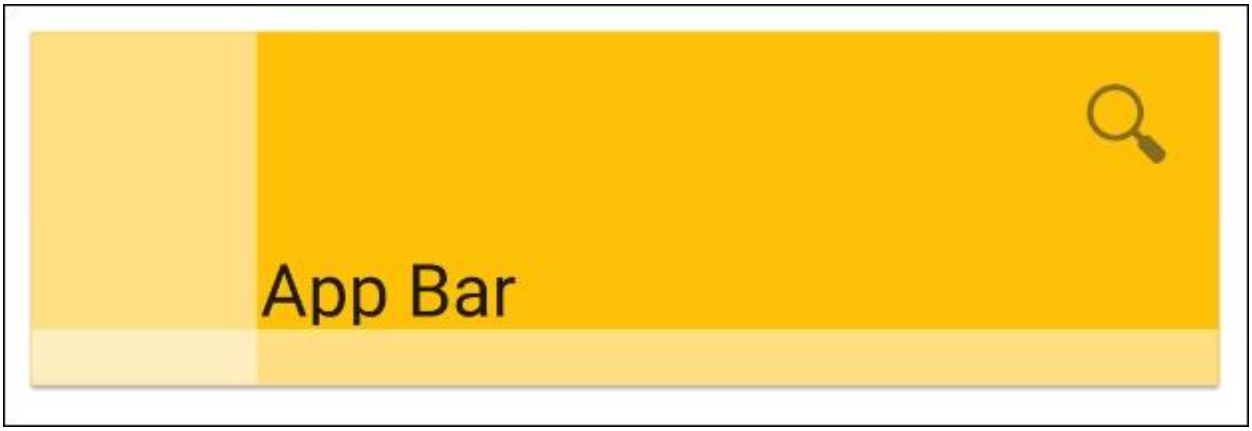


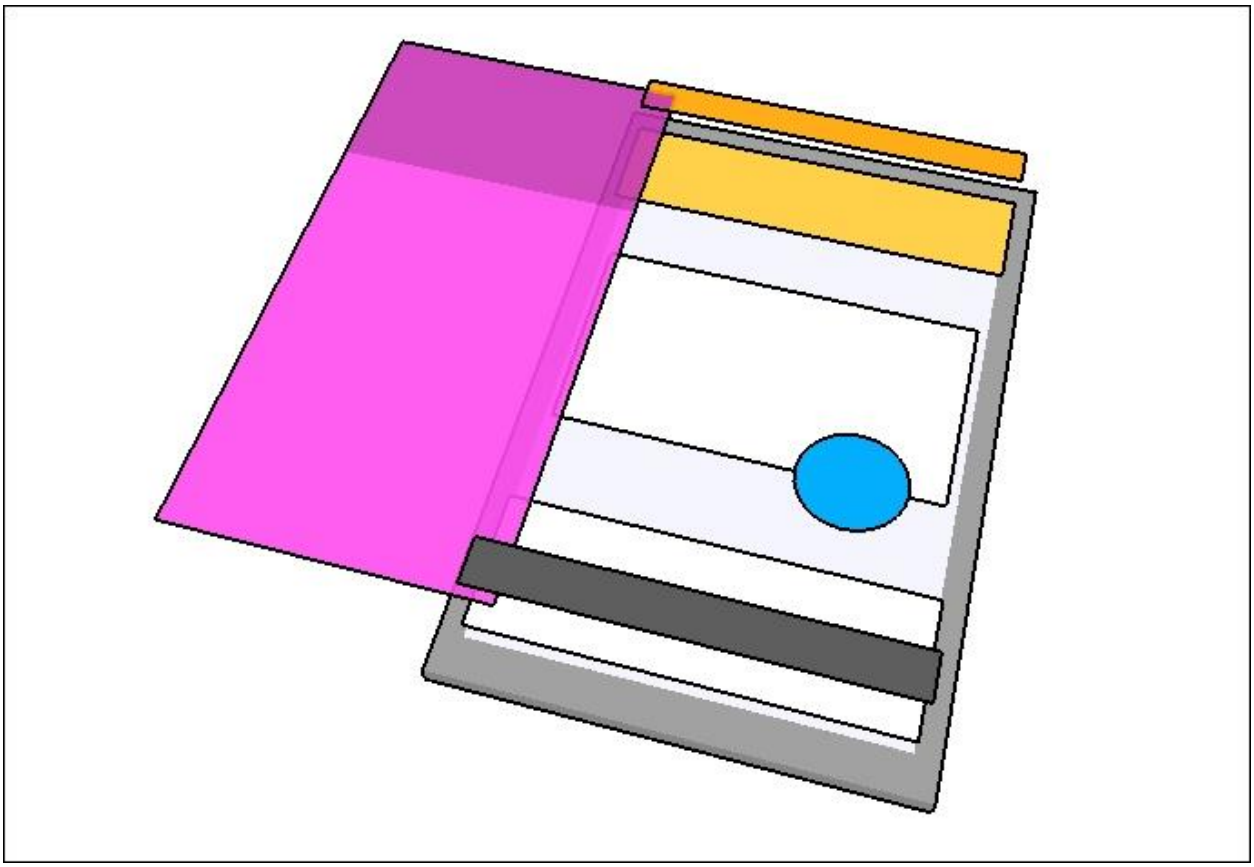



Generate Icons
Confirm Icon Path
Android Studio
Res Directory:
src\main\res
Output Directories:
main
res
drawable-mdpi
ic_action_search.png
drawable-hdpi
ic_action_search.png
drawable-xhdpi
ic_action_search.png
drawable-xxhdpi
ic_action_search.png
Previous
Next
Cancel
Finish

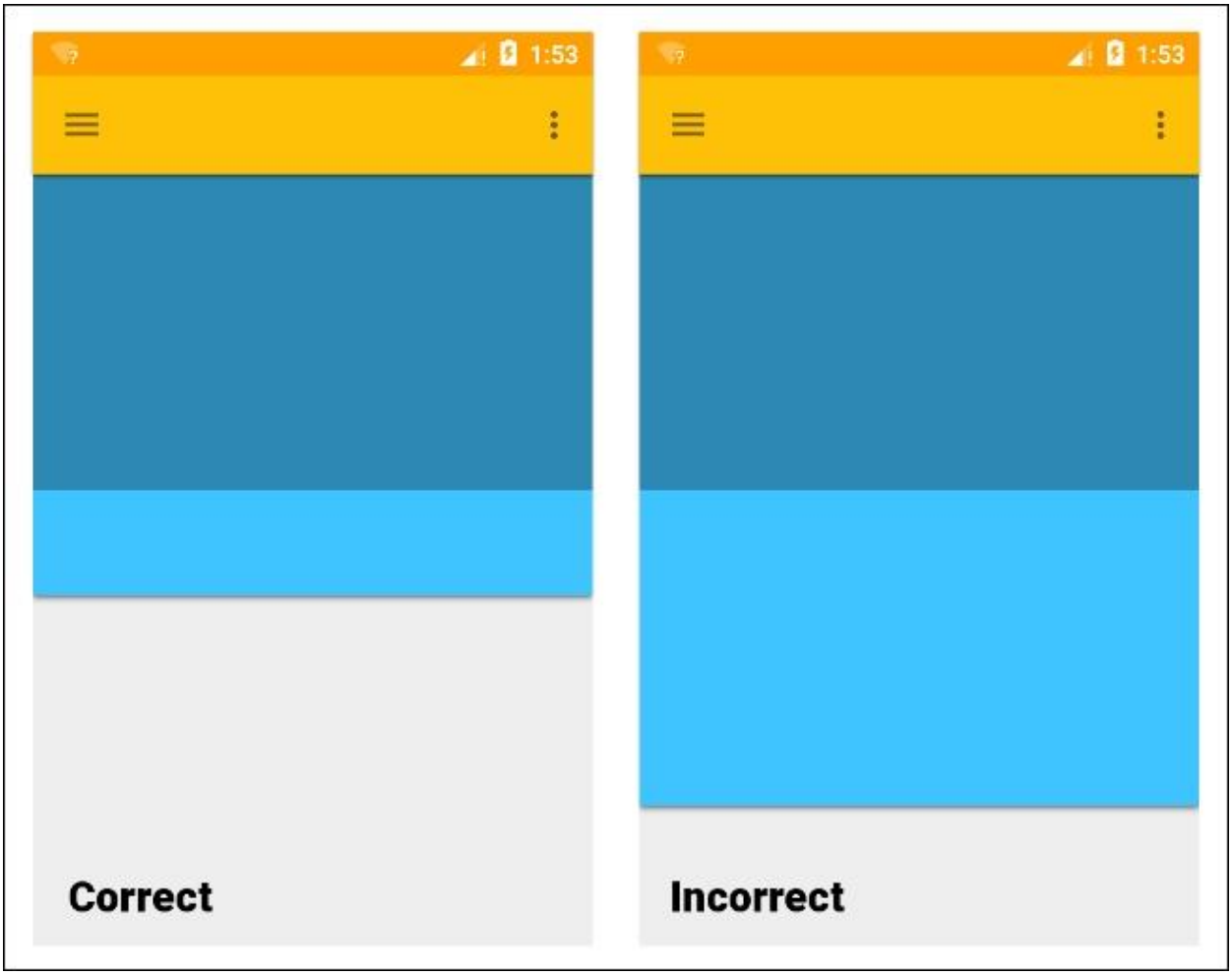

8:06
A toolbar
with a subtitle

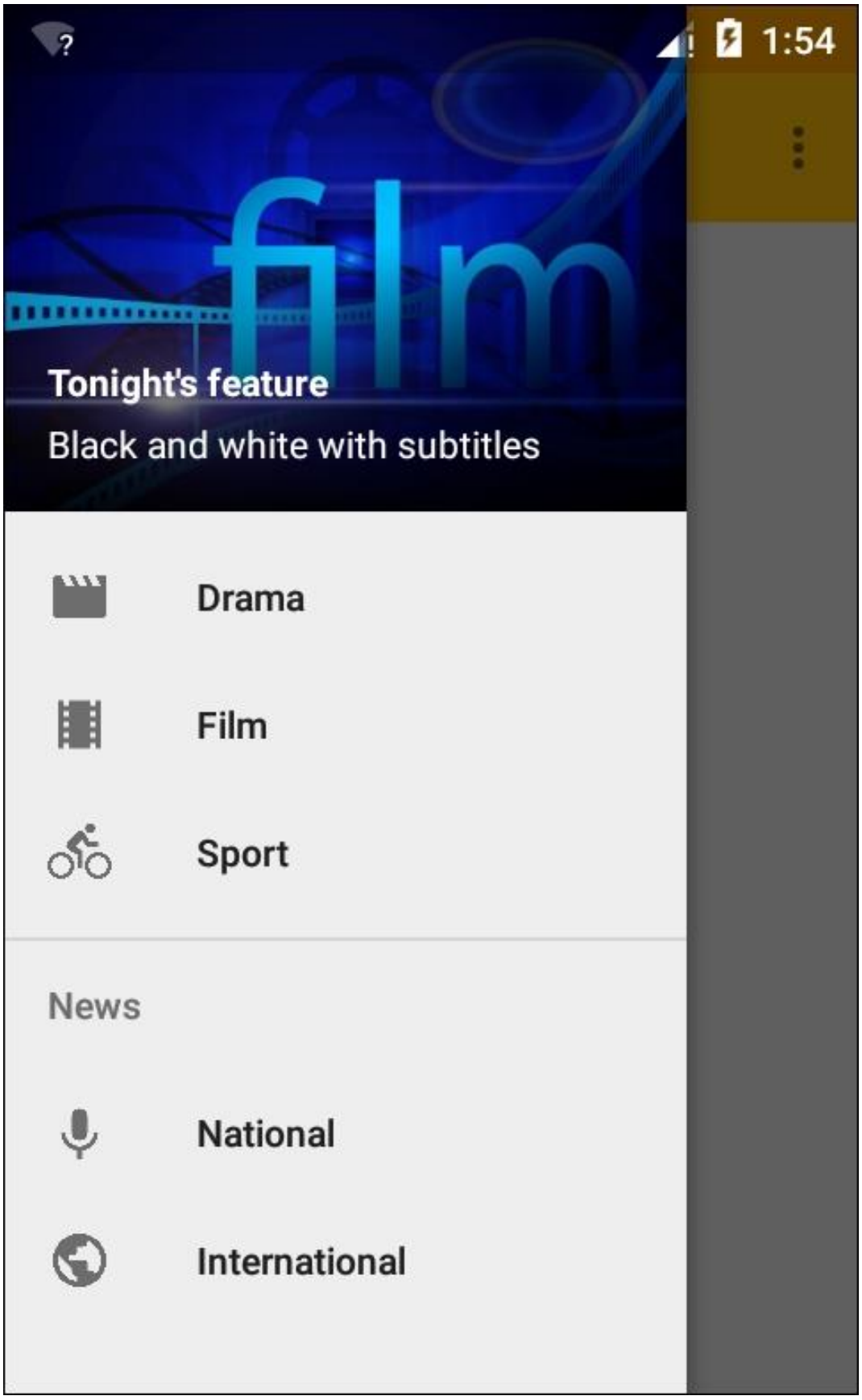
App Bar

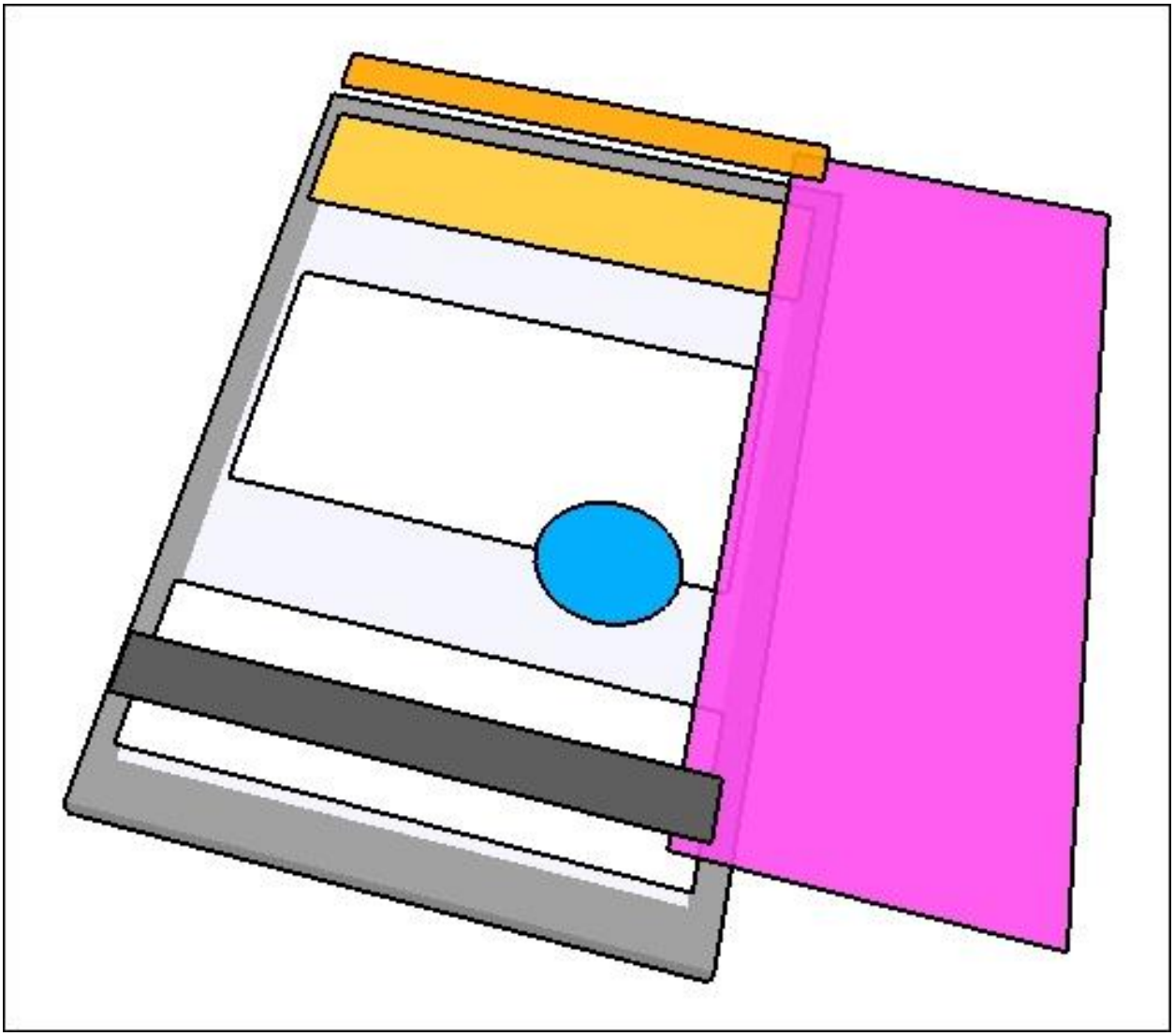

App Bar

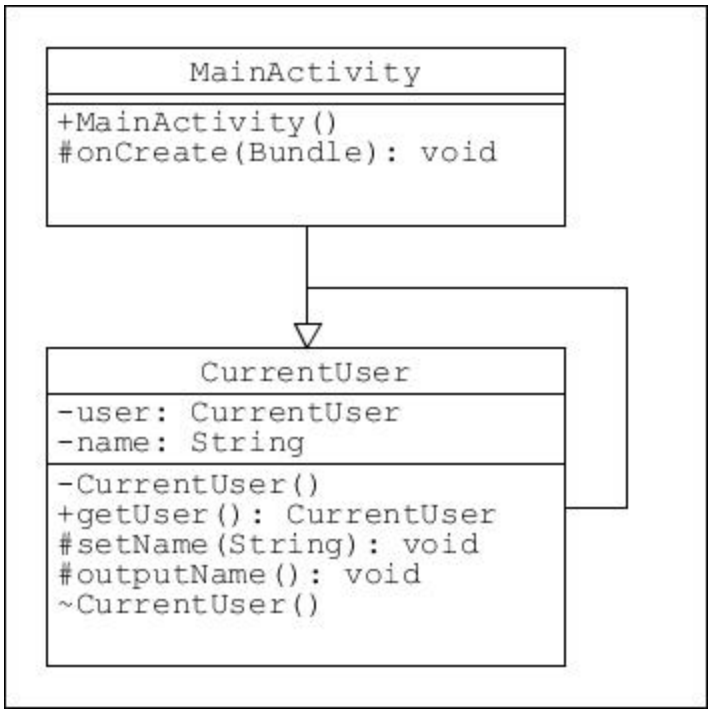


1:53
16:9
3:2
4:3
1:1

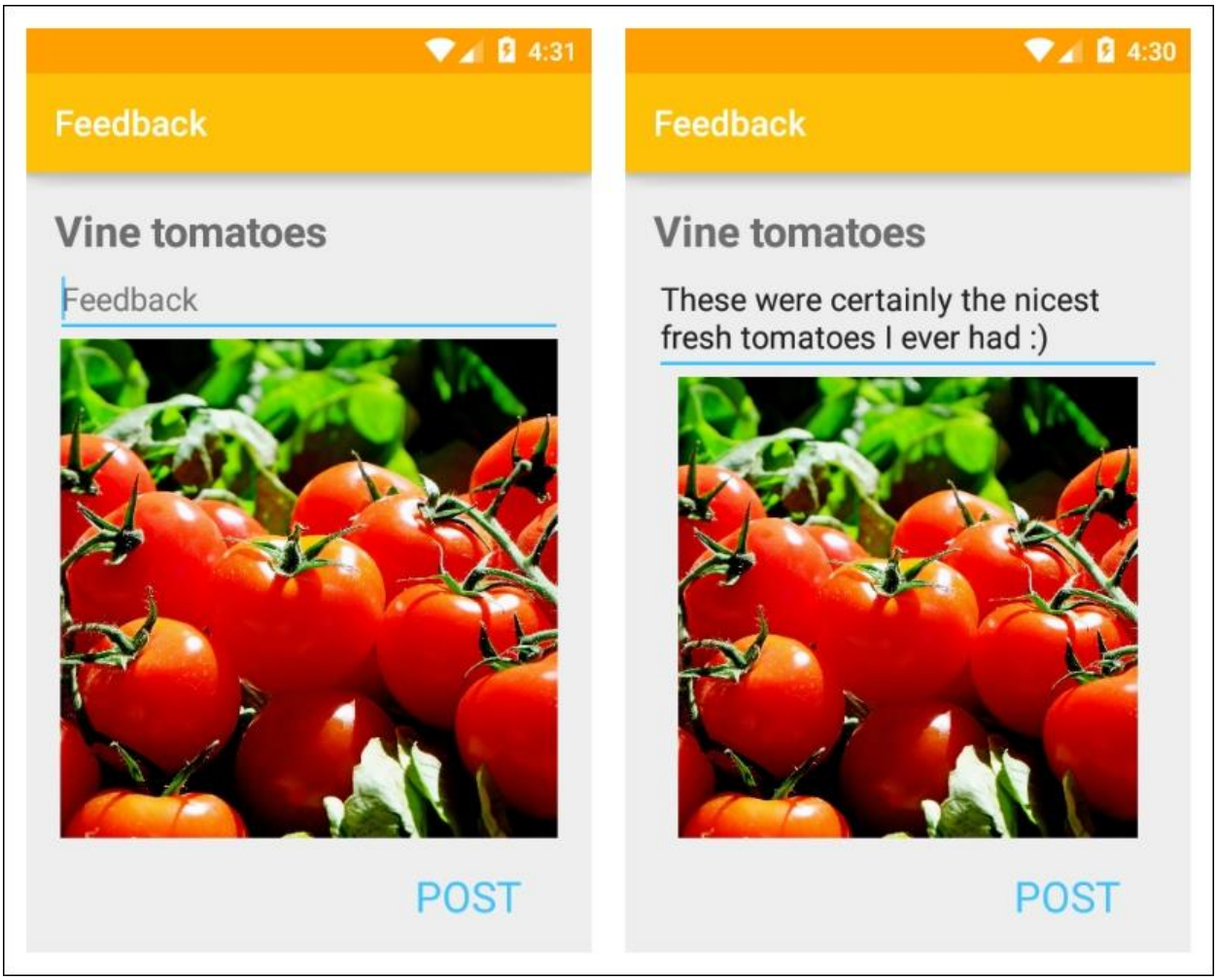

1:53
Correct
1:53
Incorrect

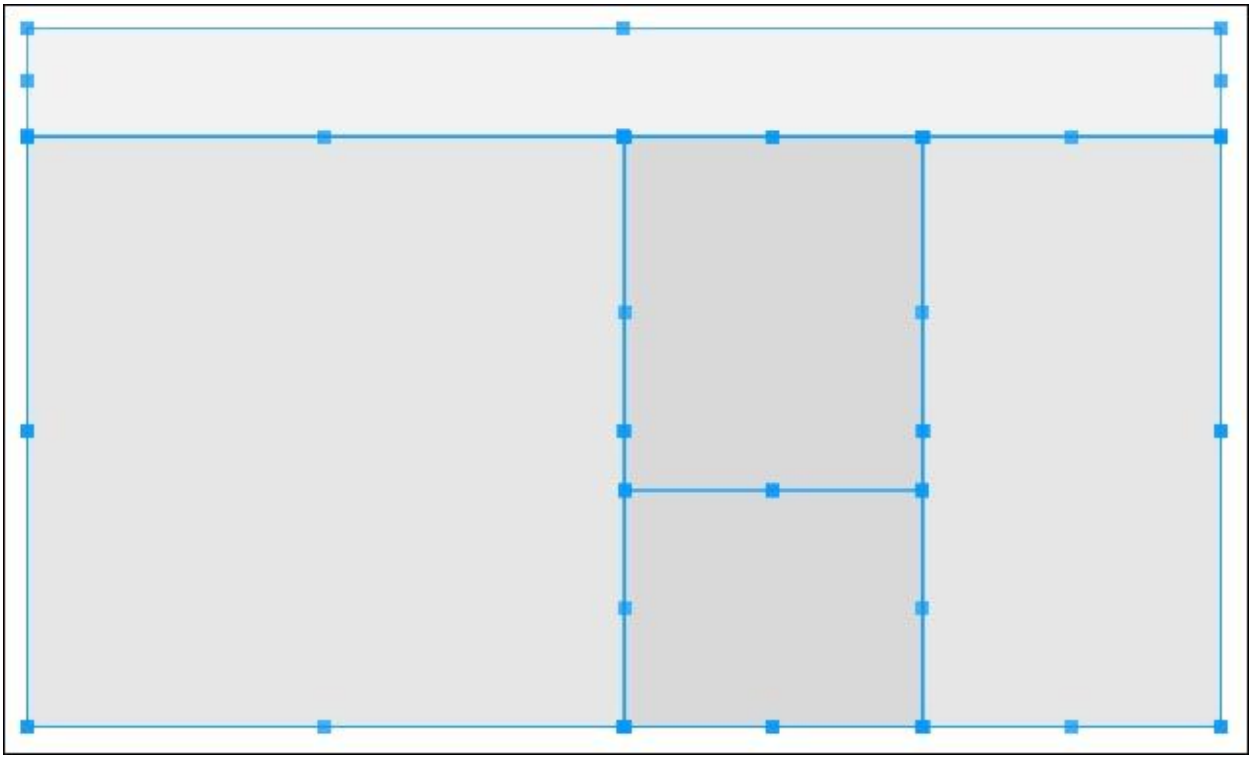

1:54
film
Tonight's feature
Black and white with subtitles
Drama
Film
Sport
News
National
International

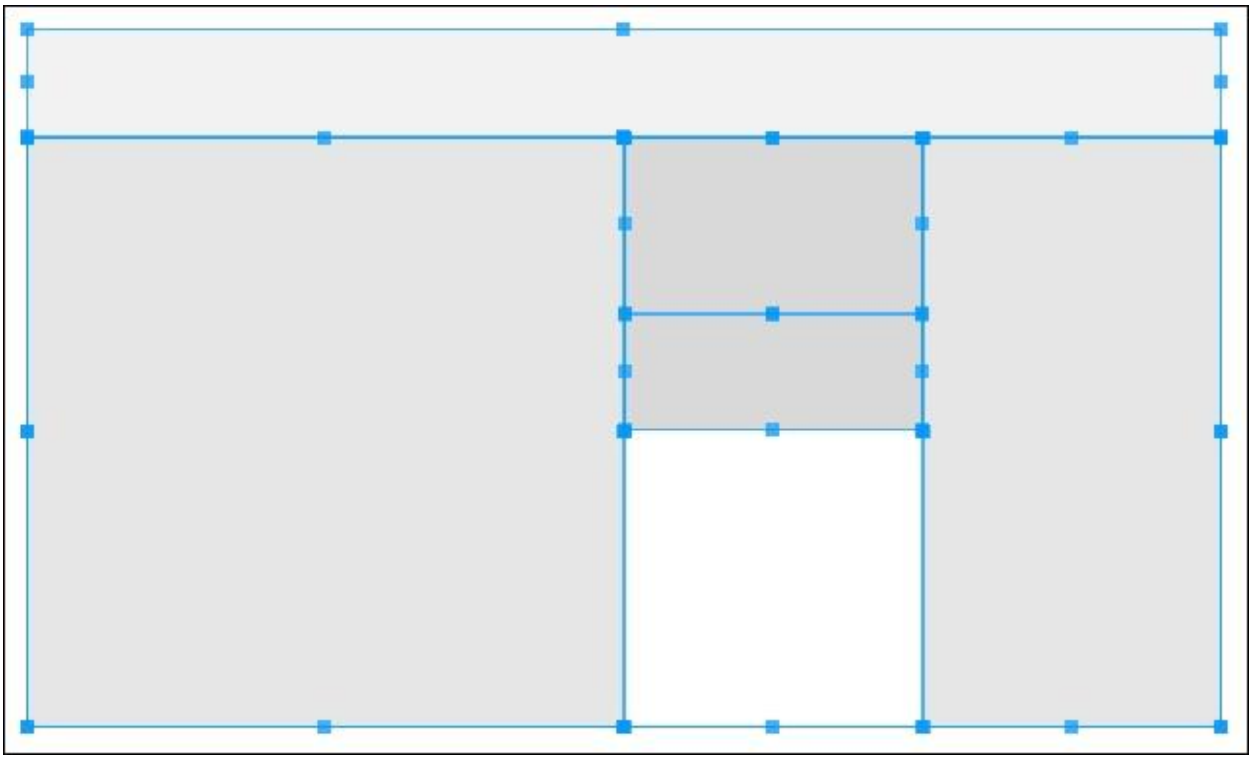


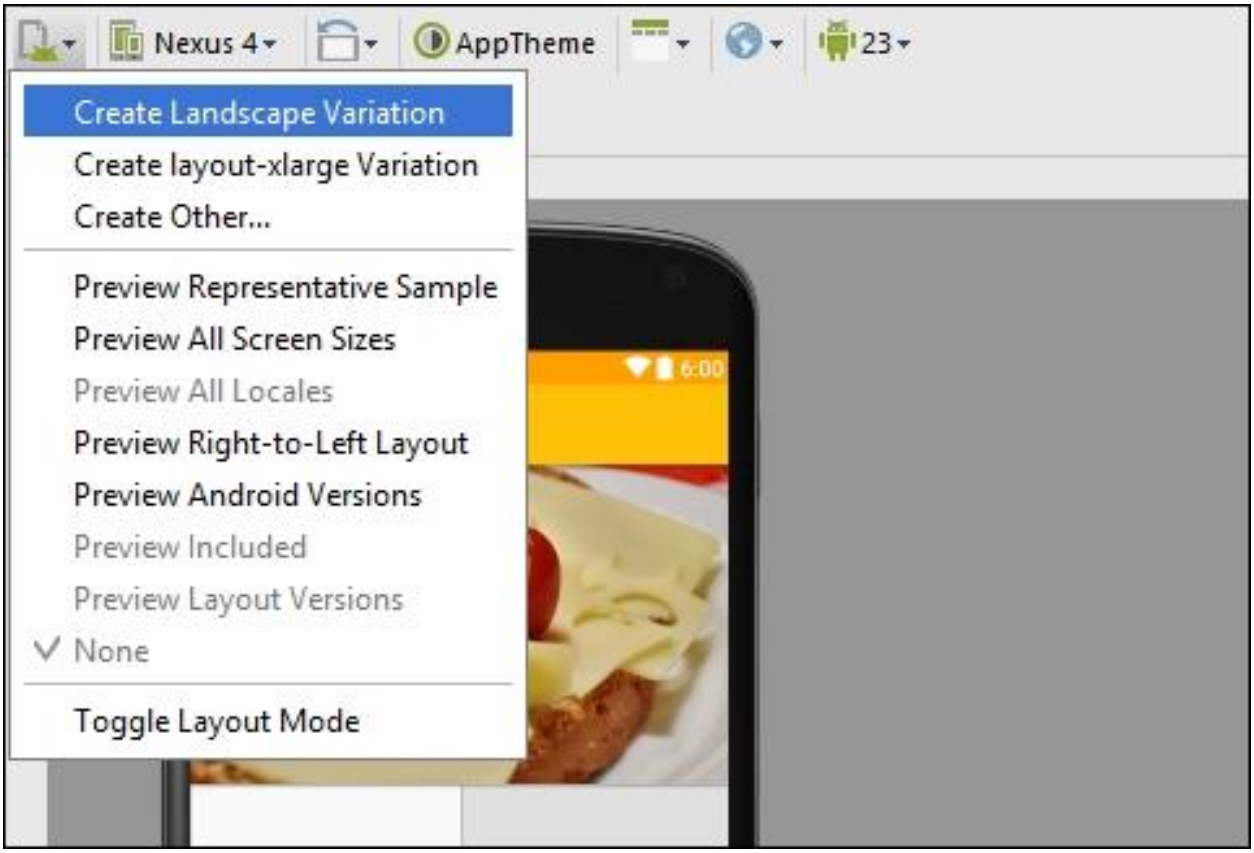

MainActivity
+MainActivity()
#onCreate(Bundle): void
CurrentUser
-user: CurrentUser
-name: String
-CurrentUser()
+getUser(): CurrentUser
#setName(String): void
#outputName(): void
~CurrentUser()

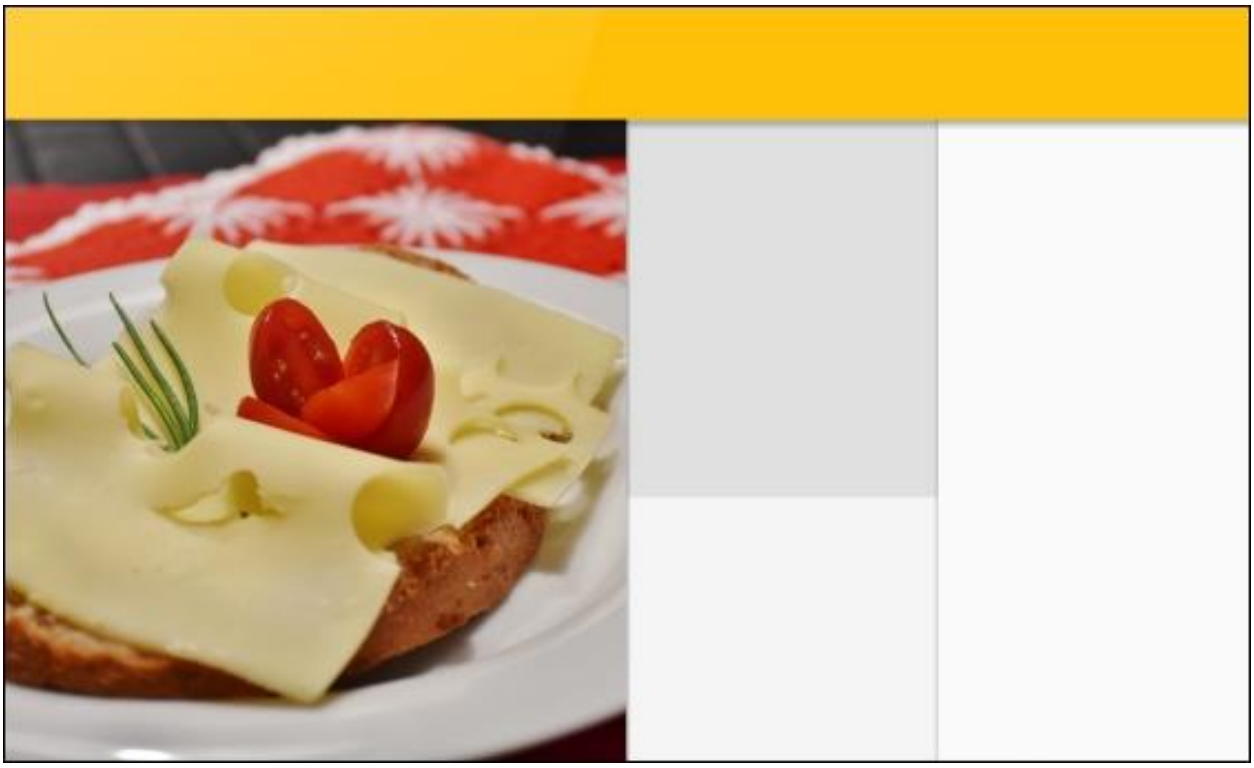
# Chapter 4: Layout Patterns

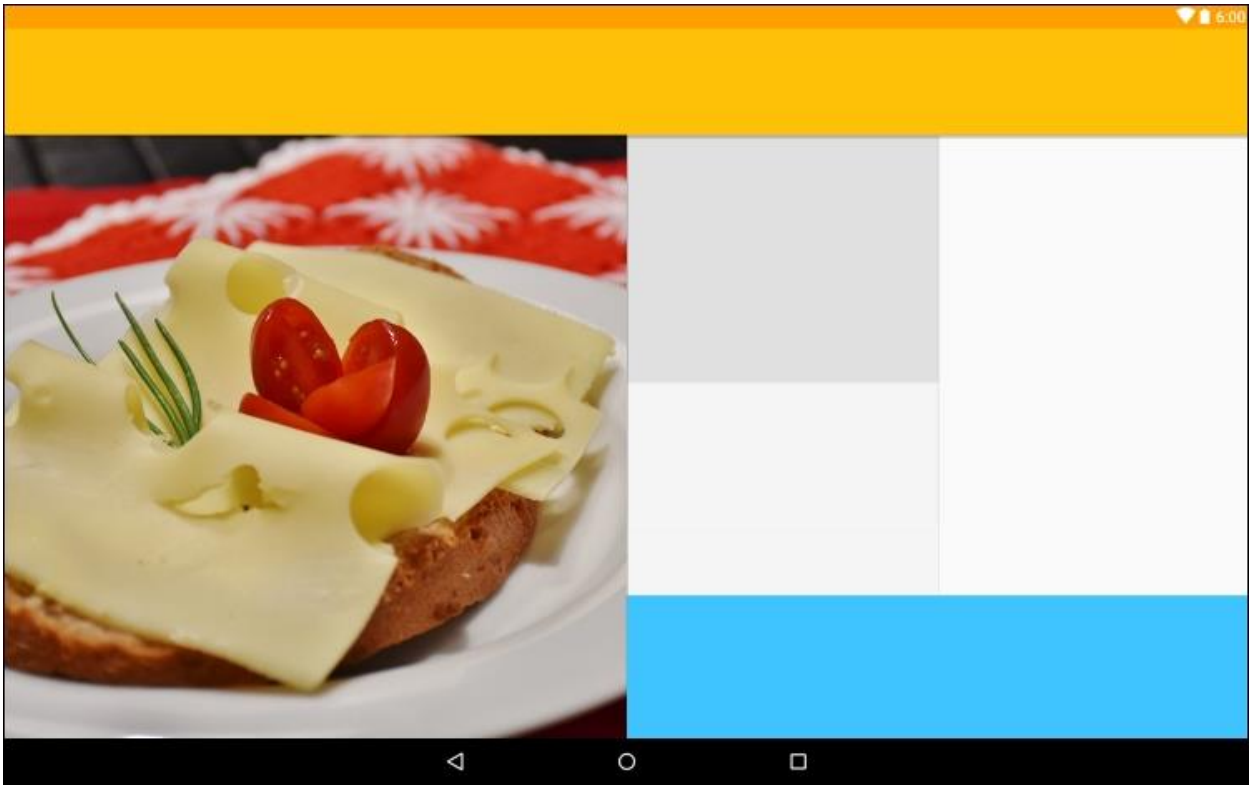







Nexus 4
AppTheme
23
Create Landscape Variation
Create layout-xlarge Variation
Create Other...
Preview Representative Sample
Preview All Screen Sizes
Preview All Locales
Preview Right-to-Left Layout
Preview Android Versions
Preview Included
Preview Layout Versions
None
Toggle Layout Mode
6:00




6:00

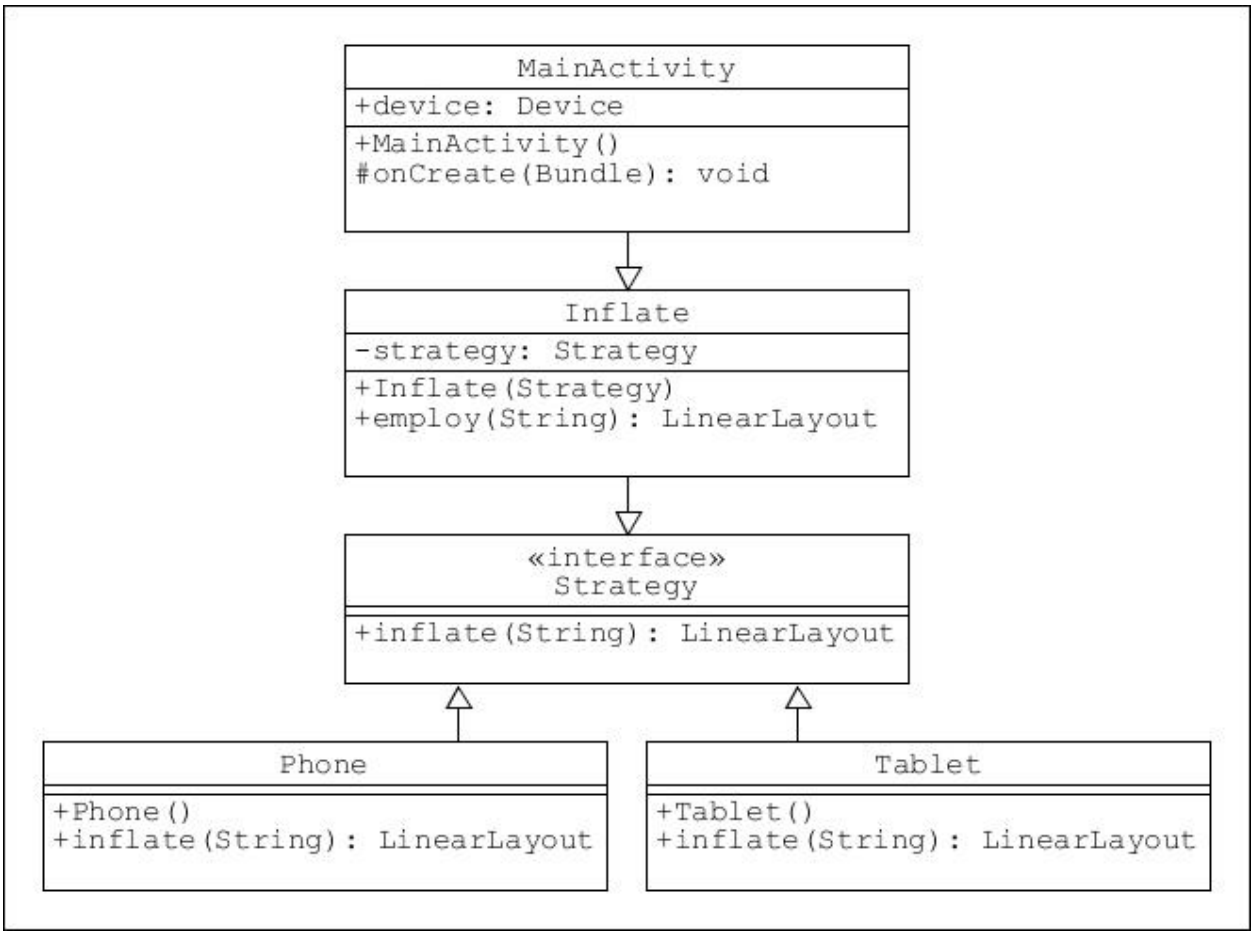
Layout Alias Example

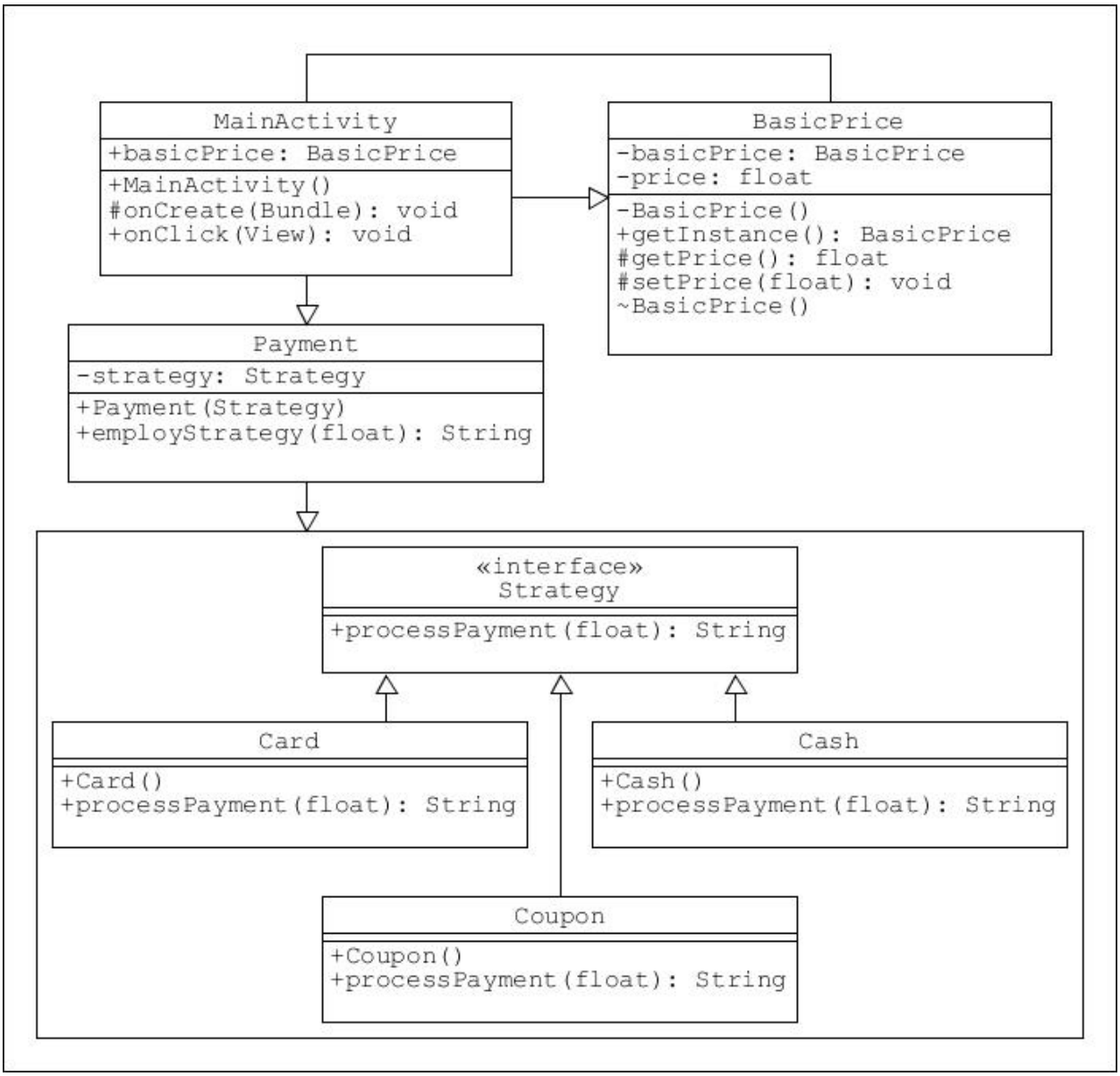
Sandwich

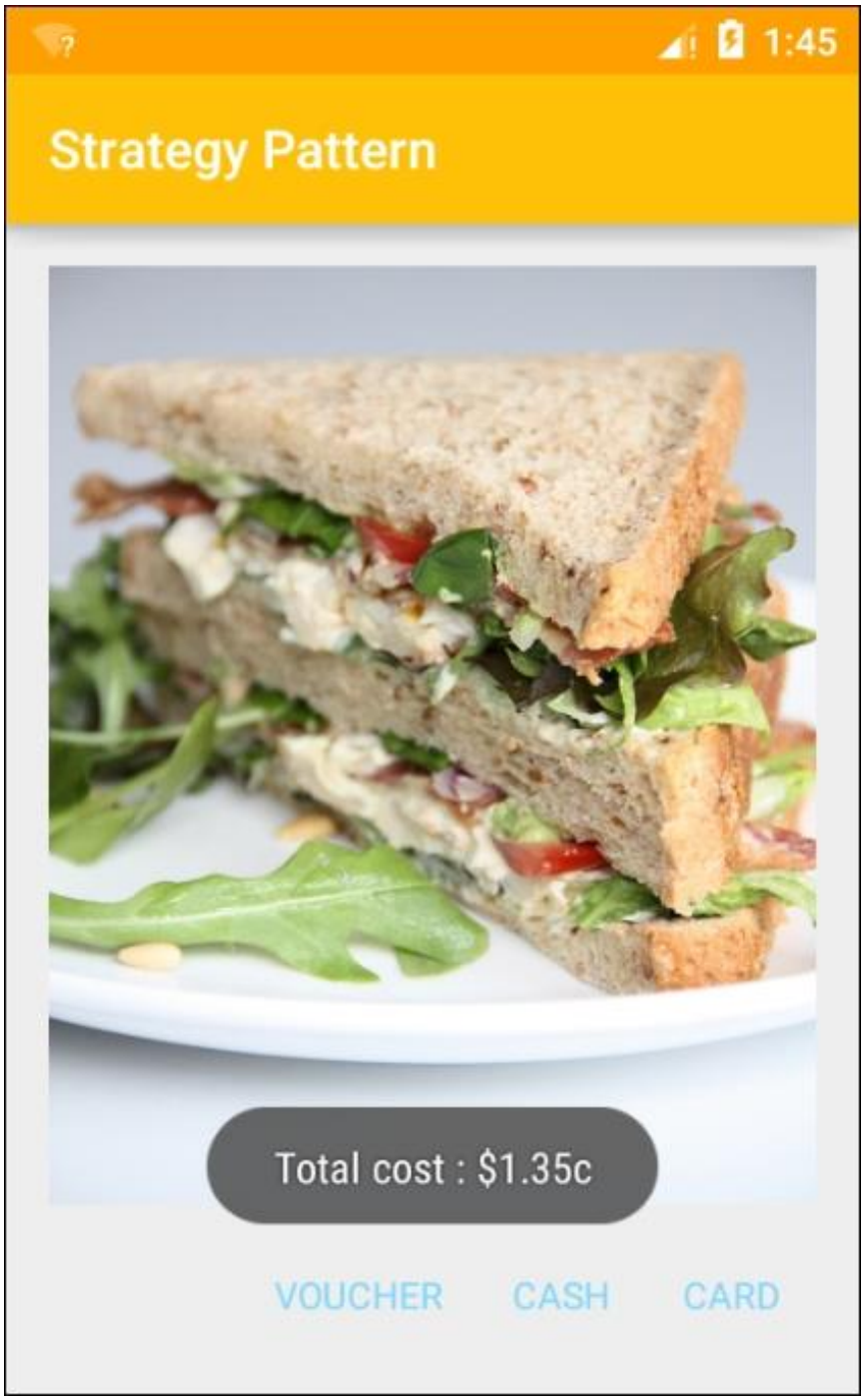
Layout Alias Example

Sandwich

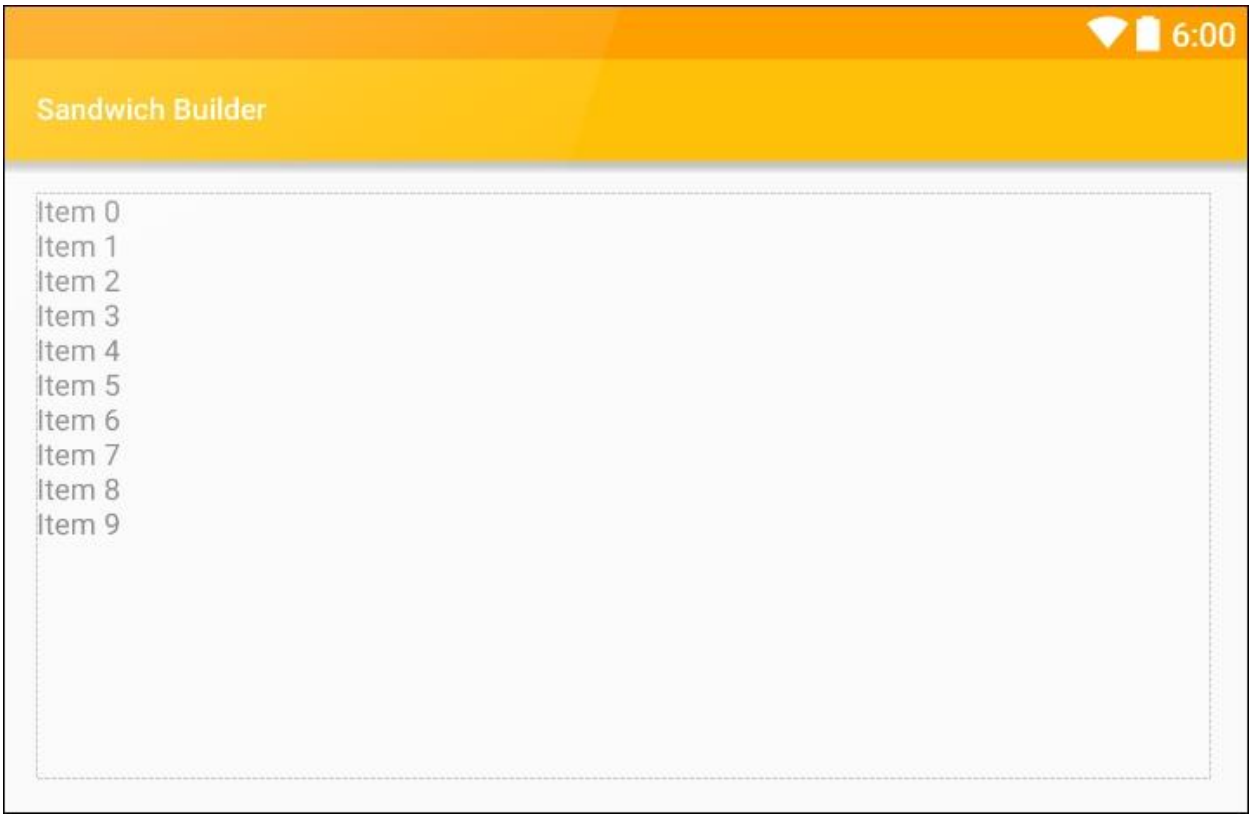
This is some content that we only show when the screen is large enough. It even works on some very old devices.

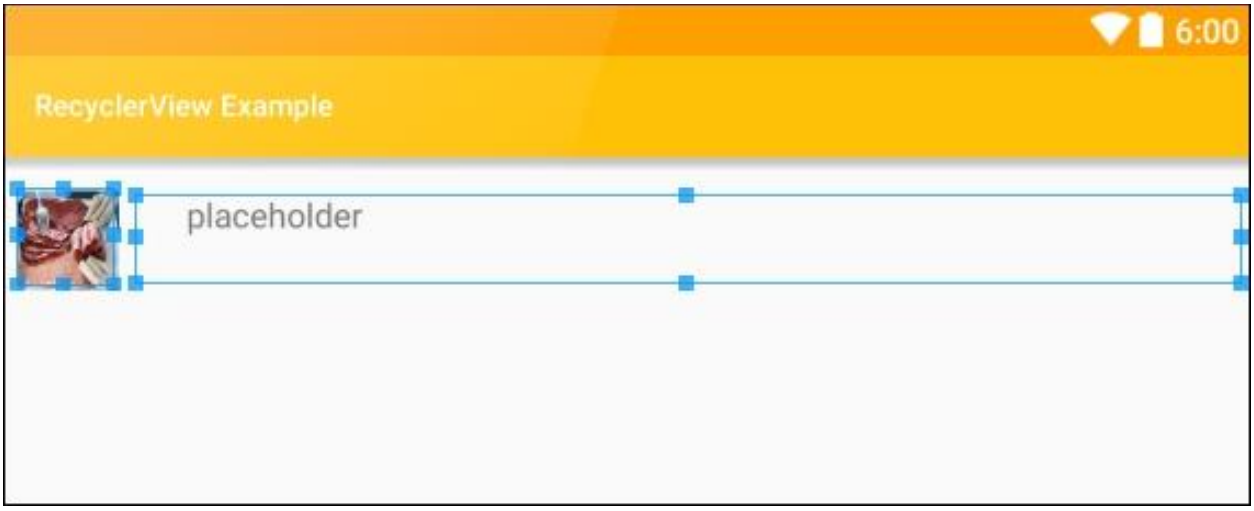

MainActivity
+device: Device
+MainActivity()
#onCreate(Bundle): void
Inflate
-strategy: Strategy
+Inflate(Strategy)
+employ(String): LinearLayout
«interface»
Strategy
+inflate(String): LinearLayout
Phone
+Phone()
+inflate(String): LinearLayout
Tablet
+Tablet()
+inflate(String): LinearLayout


MainActivity
+basicPrice: BasicPrice
+MainActivity()
#onCreate(Bundle): void
+onClick(View): void
BasicPrice
-basicPrice: BasicPrice
-price: float
-BasicPrice()
+getInstance(): BasicPrice
#getPrice(): float
#setPrice(float): void
~BasicPrice()
Payment
-strategy: Strategy
+Payment(Strategy)
+employStrategy(float): String
«interface»
Strategy
+processPayment(float): String
Card
+Card()
+processPayment(float): String
Cash
+Cash()
+processPayment(float): String
Coupon
+Coupon()
+processPayment(float): String


1:45
Strategy Pattern
Total cost : $1.35c
VOUCHER
CASH
CARD

# Chapter 5: Structural Patterns



# Display 1: Regular 34sp

## Headline: Regular 24sp

### Title: Medium 20sp

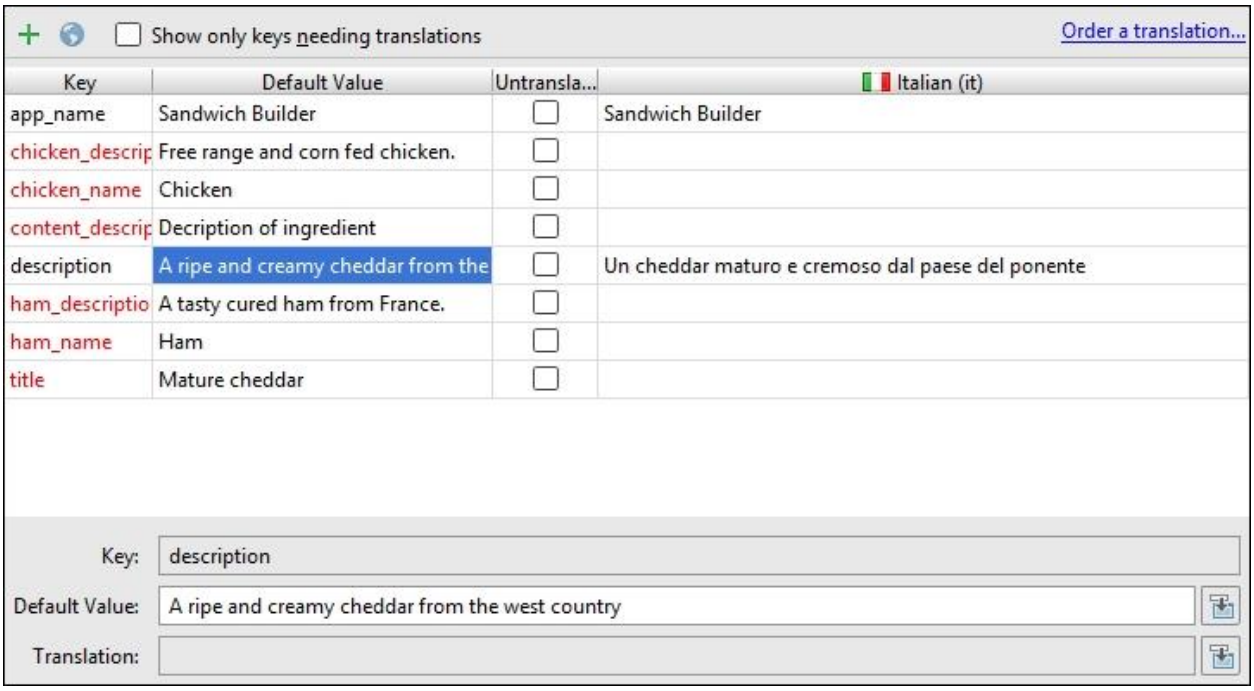
**Subhead: Regular 16sp**

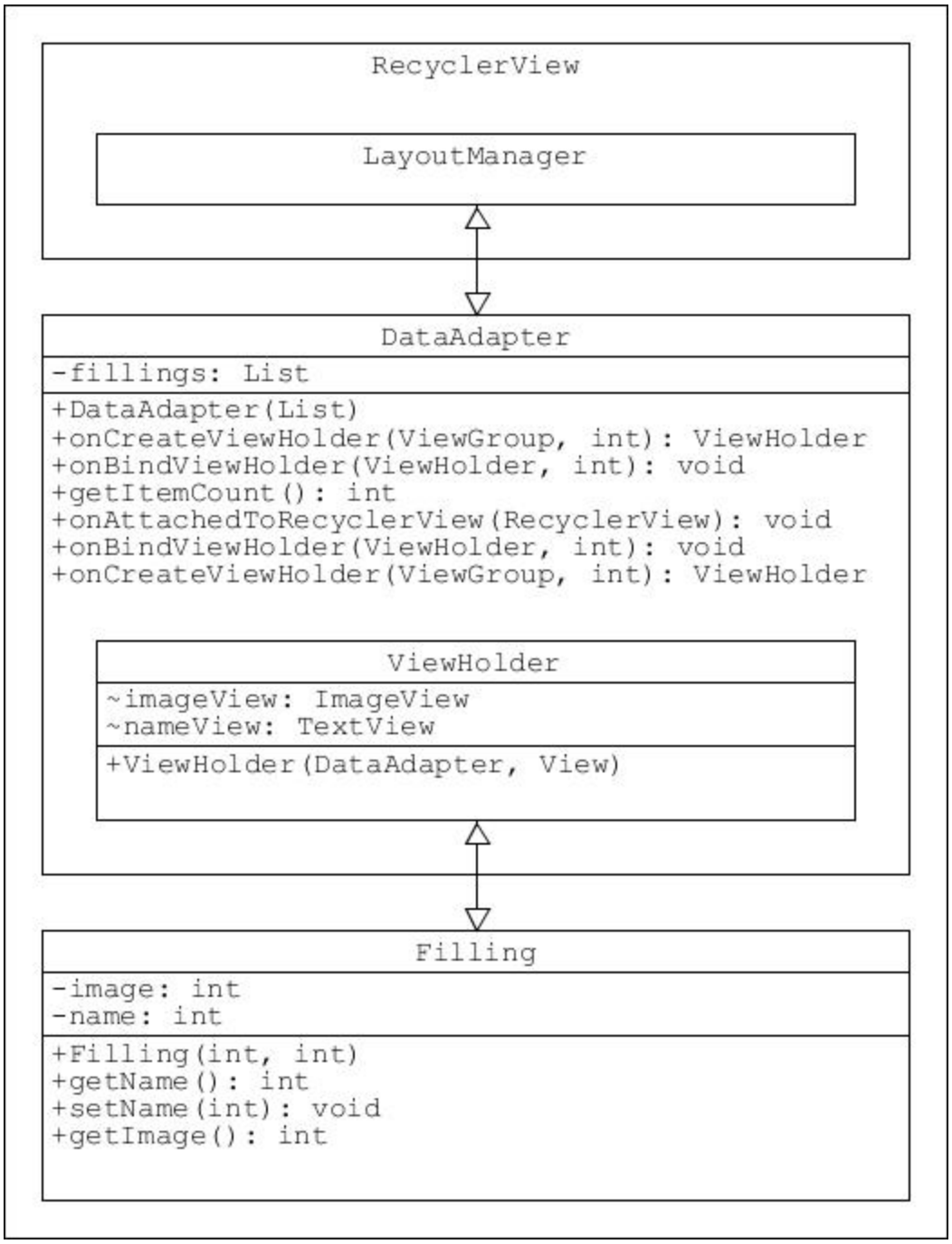
**Body 2: Medium 14sp**

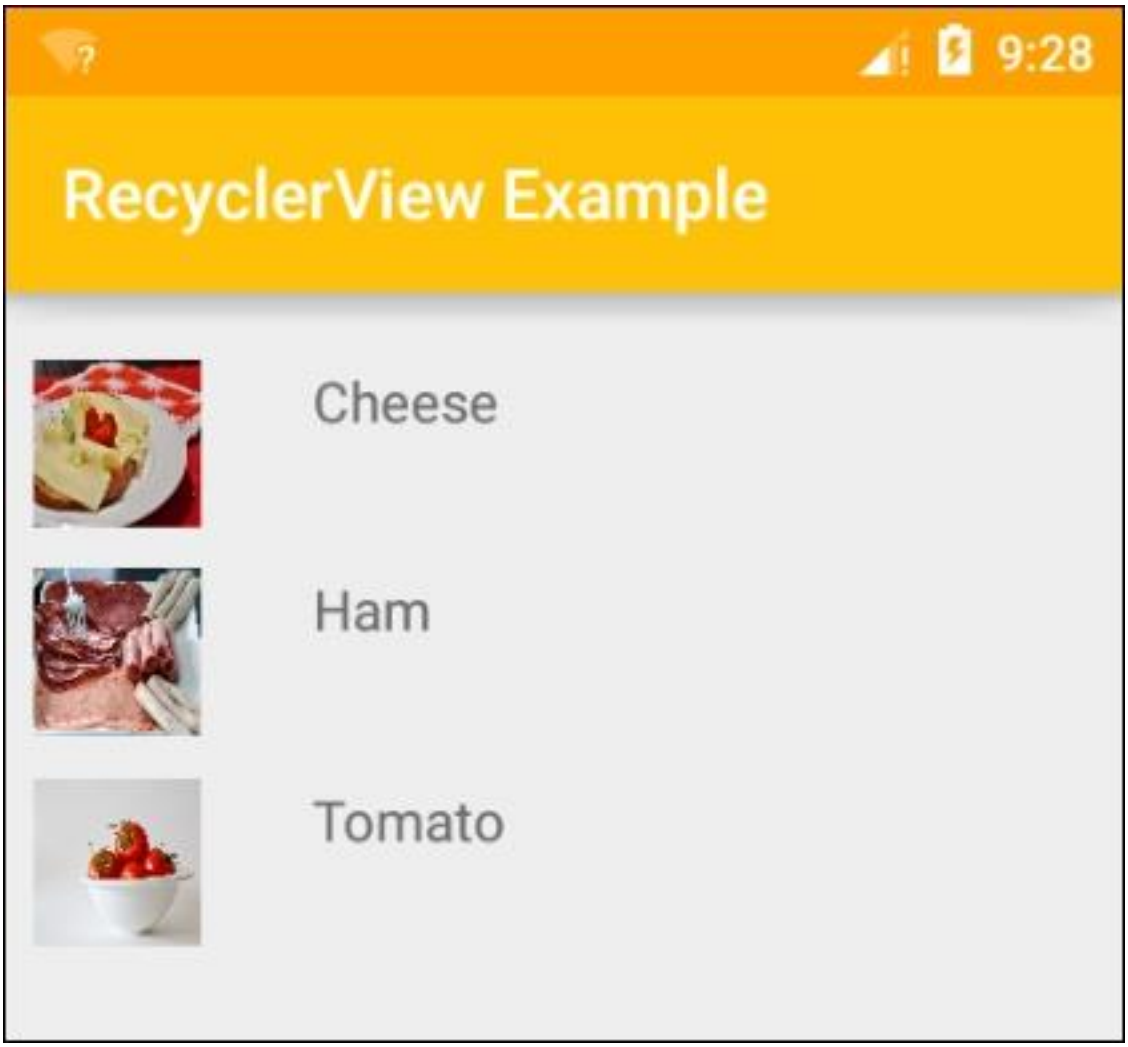
**Body 1: Regular 14sp**

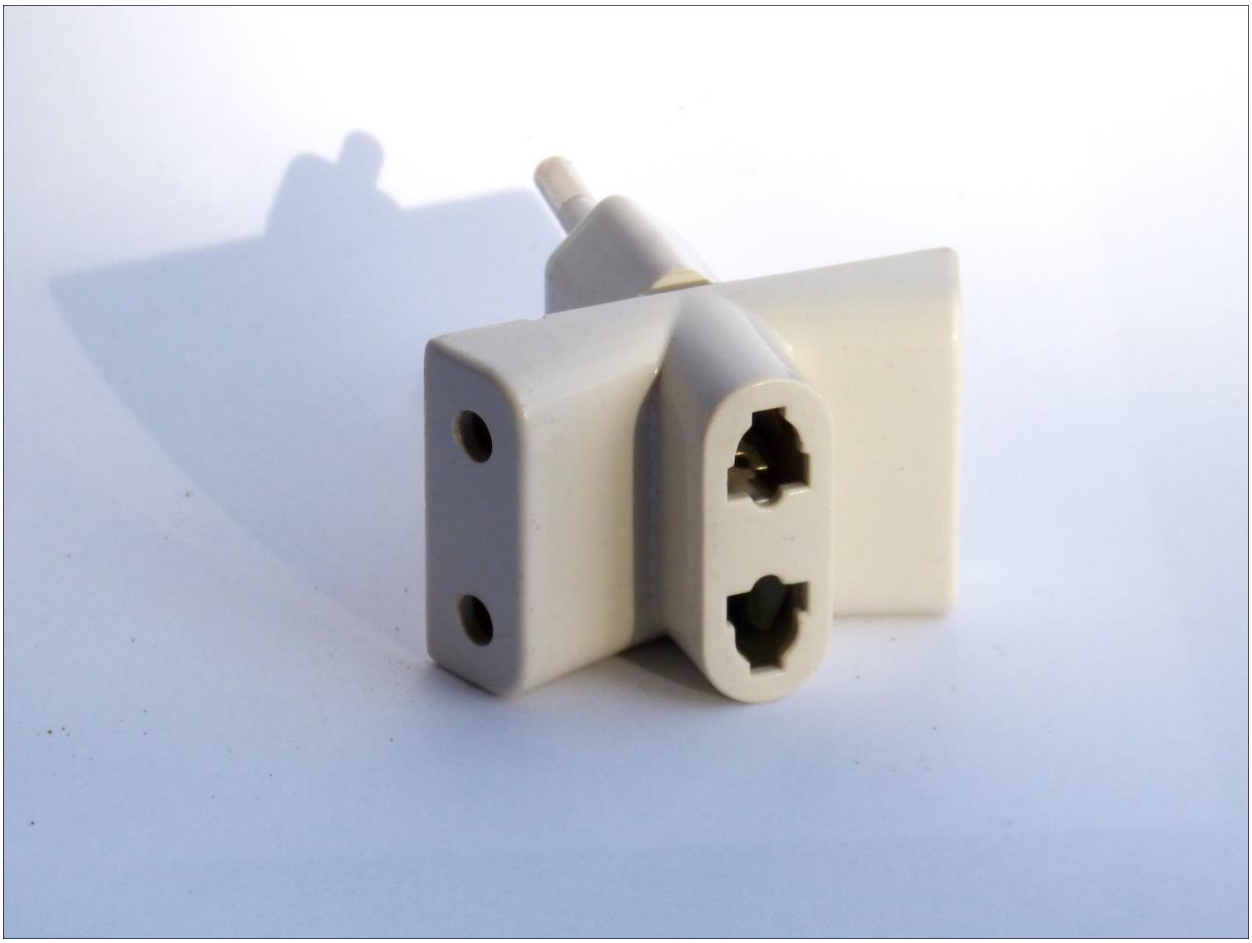
**Caption: Regular 12sp**

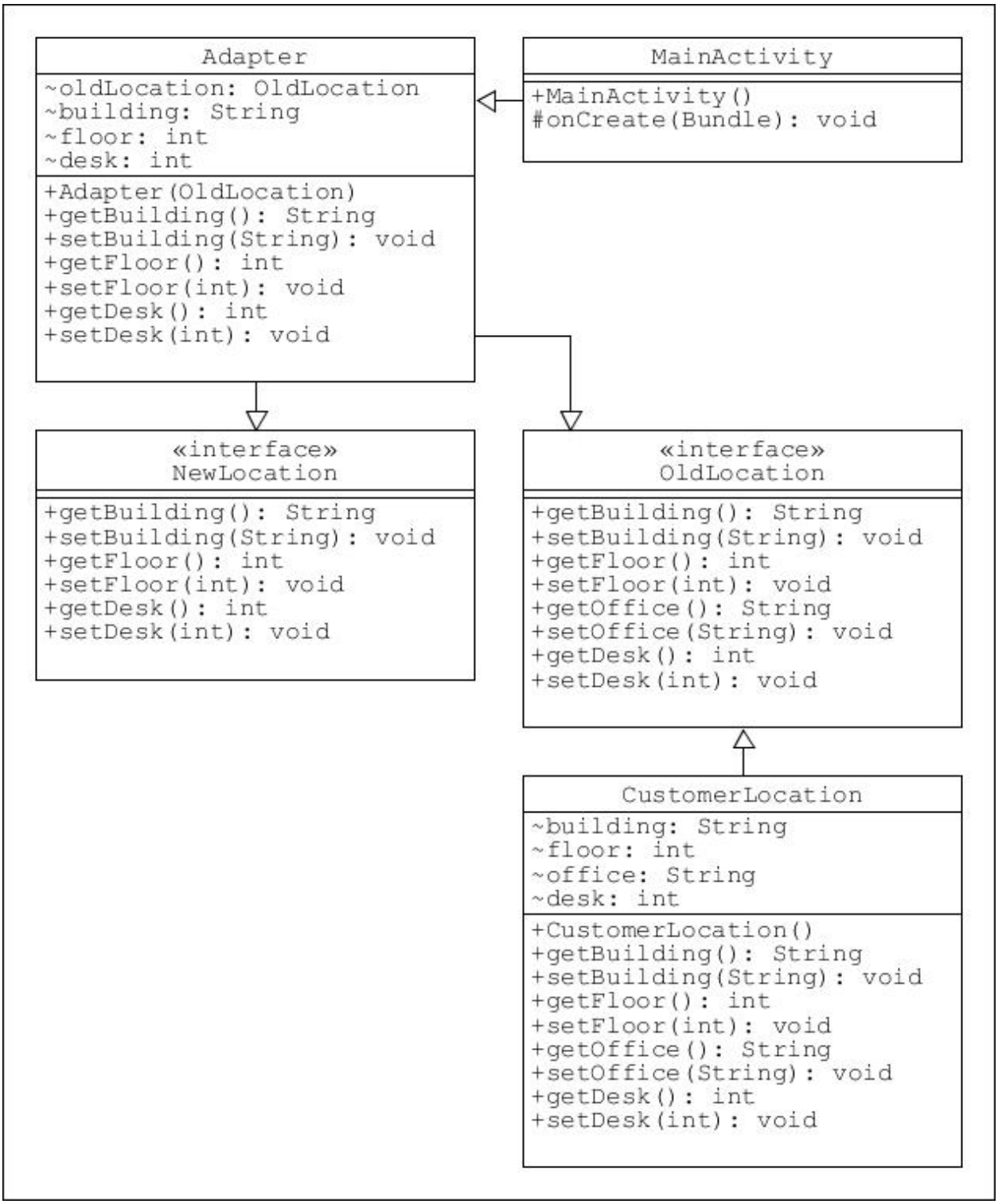
**Button: MEDIUM ALL CAPS 14sp**

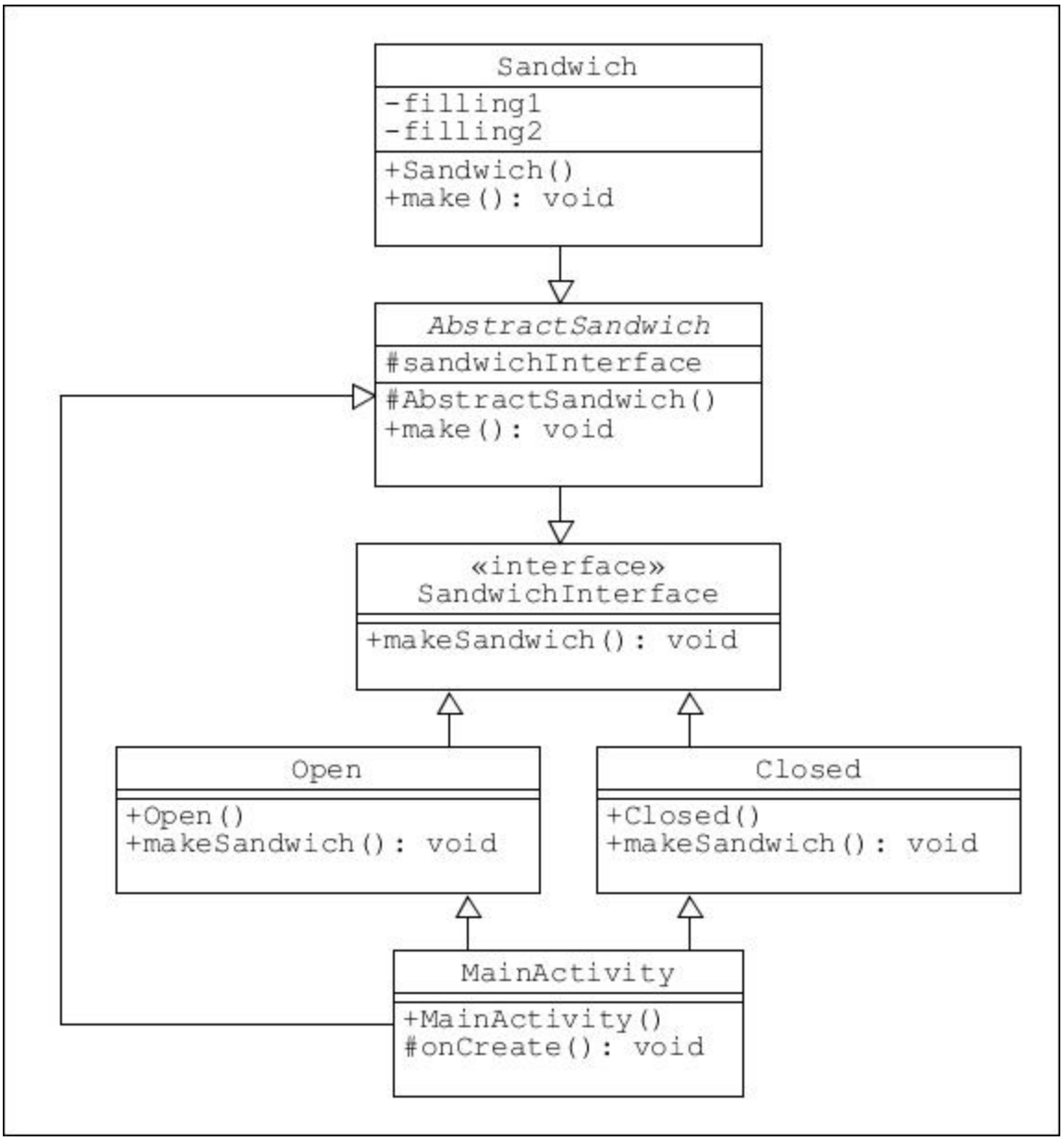


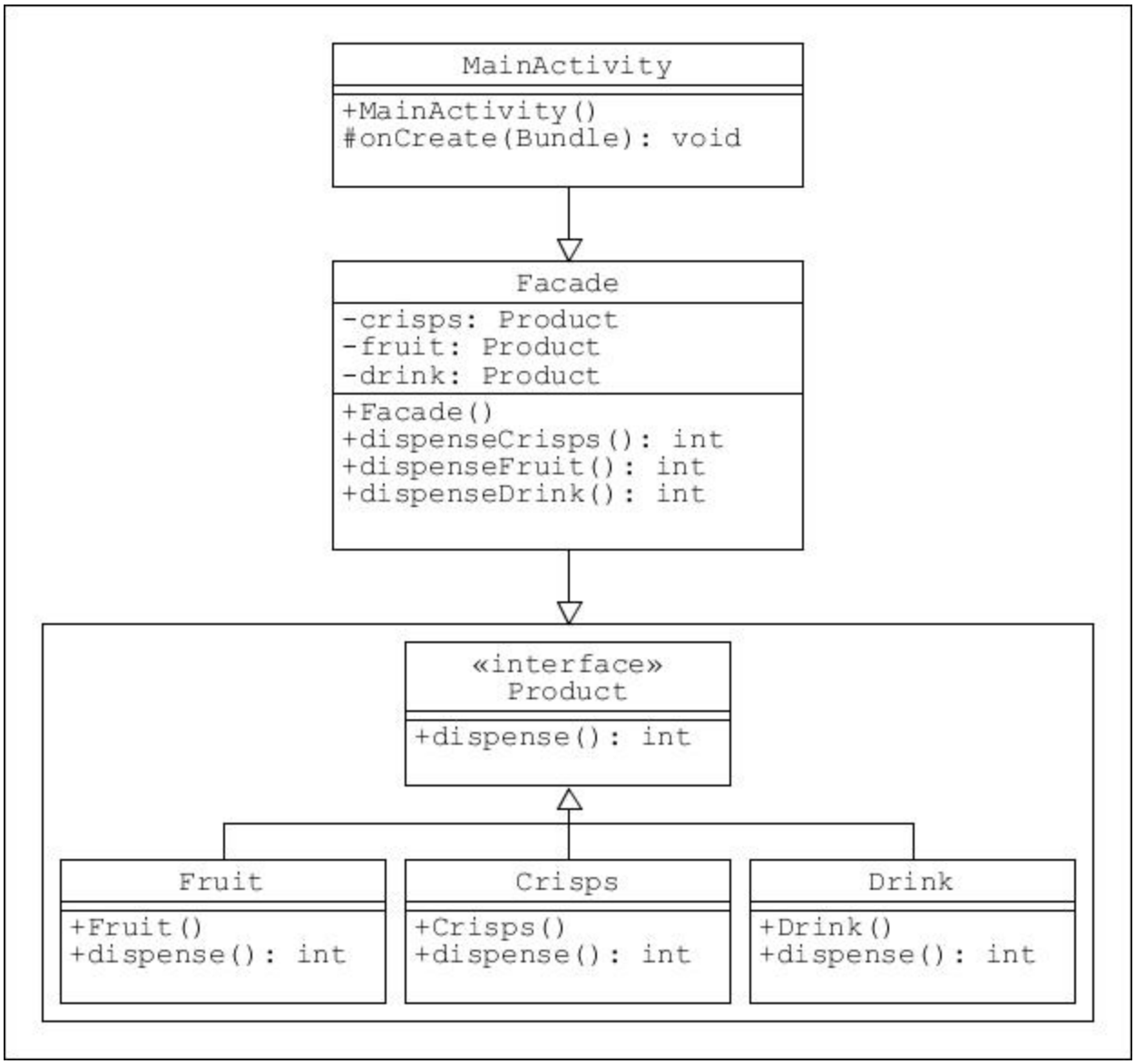

RecyclerView
LayoutManager
DataAdapter
-fillings: List
+DataAdapter(List)
+onCreateViewHolder(ViewGroup, int): ViewHolder
+onBindViewHolder(ViewHolder, int): void
+getItemCount(): int
+onAttachedToRecyclerView(RecyclerView): void
+onBindViewHolder(ViewHolder, int): void
+onCreateViewHolder(ViewGroup, int): ViewHolder
ViewHolder
~imageView: ImageView
~nameView: TextView
+ViewHolder(DataAdapter, View)
Filling
-image: int
-name: int
+Filling(int, int)
+getName(): int
+setName(int): void
+getImage(): int


9:28
RecyclerView Example
Cheese
Ham
Tomato

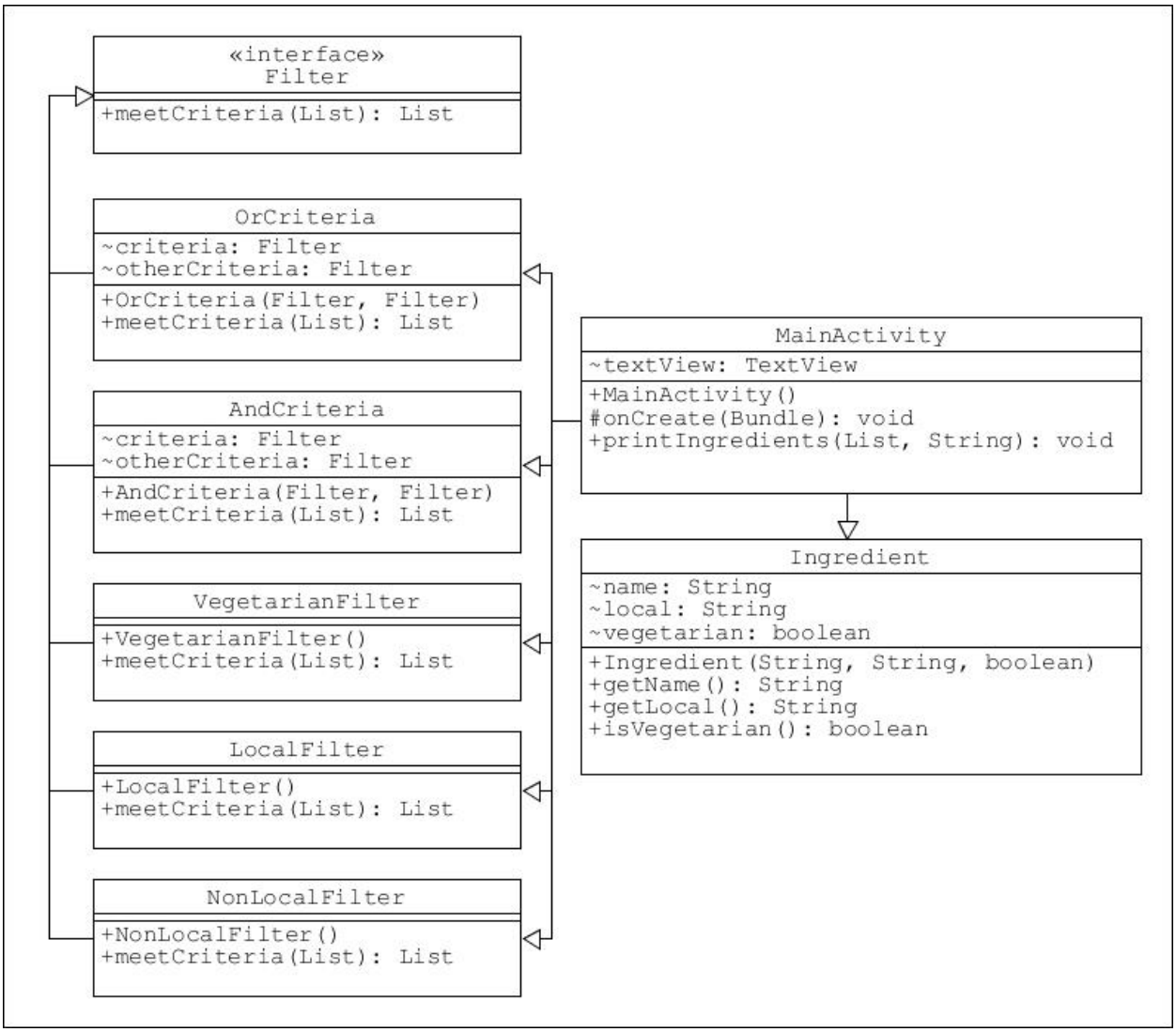



Adapter
~oldLocation: OldLocation
~building: String
~floor: int
~desk: int
+Adapter(OldLocation)
+getBuilding(): String
+setBuilding(String): void
+getFloor(): int
+setFloor(int): void
+getDesk(): int
+setDesk(int): void
MainActivity
+MainActivity()
#onCreate(Bundle): void
«interface»
NewLocation
+getBuilding(): String
+setBuilding(String): void
+getFloor(): int
+setFloor(int): void
+getDesk(): int
+setDesk(int): void
«interface»
OldLocation
+getBuilding(): String
+setBuilding(String): void
+getFloor(): int
+setFloor(int): void
+getOffice(): String
+setOffice(String): void
+getDesk(): int
+setDesk(int): void
CustomerLocation
~building: String
~floor: int
~office: String
~desk: int
+CustomerLocation()
+getBuilding(): String
+setBuilding(String): void
+getFloor(): int
+setFloor(int): void
+getOffice(): String
+setOffice(String): void
+getDesk(): int
+setDesk(int): void

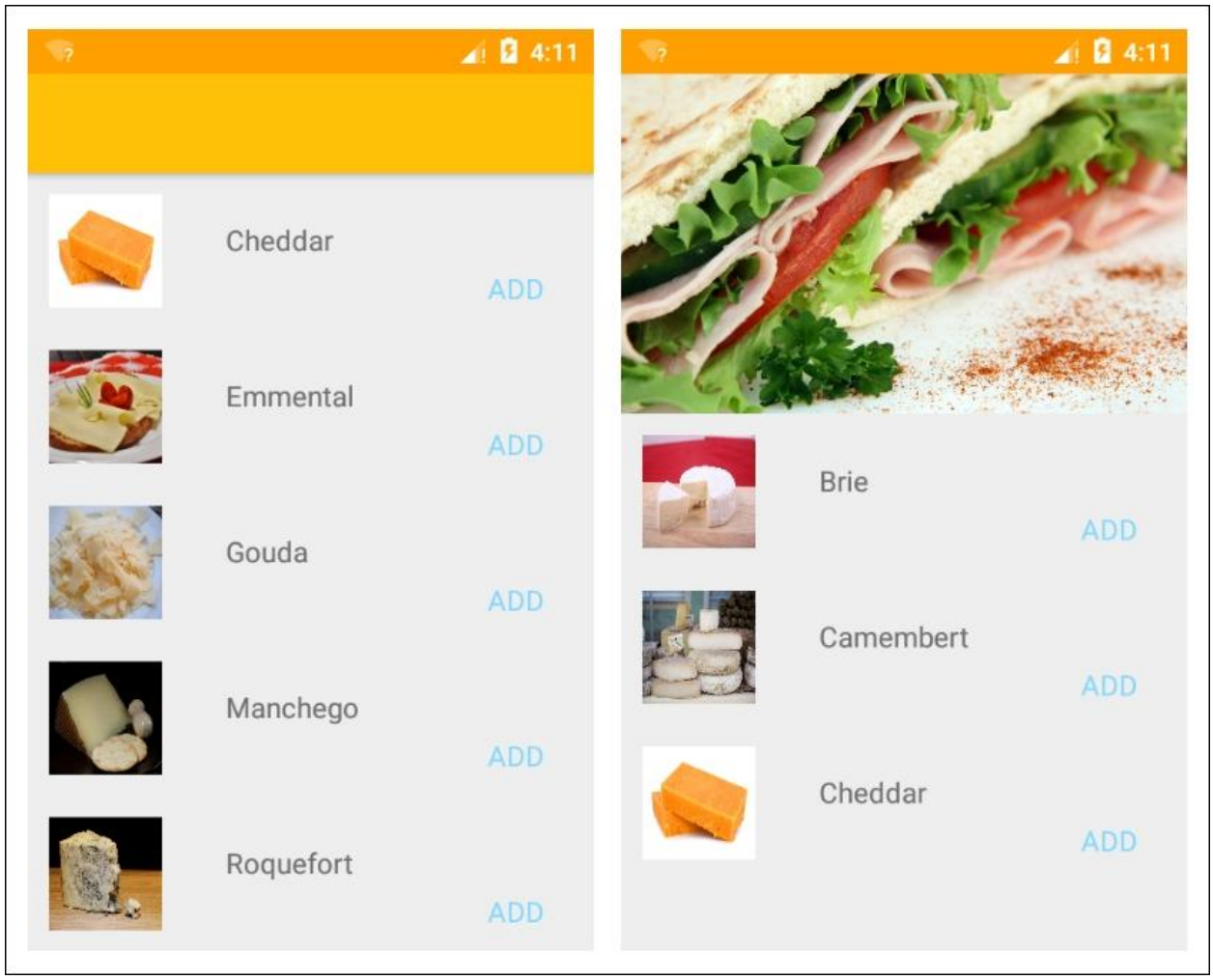

Sandwich
-filling1
-filling2
+Sandwich()
+make(): void
AbstractSandwich
#sandwichInterface
#AbstractSandwich()
+make(): void
«interface»
SandwichInterface
+makeSandwich(): void
Open
+Open()
+makeSandwich(): void
Closed
+Closed()
+makeSandwich(): void
MainActivity
+MainActivity()
#onCreate(): void


MainActivity
+MainActivity()
#onCreate(Bundle): void
Facade
-crisps: Product
-fruit: Product
-drink: Product
+Facade()
+dispenseCrisps(): int
+dispenseFruit(): int
+dispenseDrink(): int
«interface»
Product
+dispense(): int
Fruit
+Fruit()
+dispense(): int
Crisps
+Crisps()
+dispense(): int
Drink
+Drink()
+dispense(): int

5:47
Criteria Pattern
LOCAL:
Cheddar, locally produced
Turkey, locally produced
NOT LOCAL:
Ham, Cheshire
Tomato, Kent
VEGETARIAN:
Cheddar, locally produced
Tomato, Kent
LOCAL VEGETARIAN:
Cheddar, locally produced
ENVIRONMENTALLY FRIENDLY:
Cheddar, locally produced
Turkey, locally produced
Tomato, Kent


«interface»
Filter
+meetCriteria(List): List
OrCriteria
~criteria: Filter
~otherCriteria: Filter
+OrCriteria(Filter, Filter)
+meetCriteria(List): List
AndCriteria
~criteria: Filter
~otherCriteria: Filter
+AndCriteria(Filter, Filter)
+meetCriteria(List): List
VegetarianFilter
+VegetarianFilter()
+meetCriteria(List): List
LocalFilter
+LocalFilter()
+meetCriteria(List): List
NonLocalFilter
+NonLocalFilter()
+meetCriteria(List): List
MainActivity
~textView: TextView
+MainActivity()
#onCreate(Bundle): void
+printIngredients(List, String): void
Ingredient
~name: String
~local: String
~vegetarian: boolean
+Ingredient(String, String, boolean)
+getName(): String
+getLocal(): String
+isVegetarian(): boolean

# Chapter 6: Activating Patterns



4:25

6:00
This is some cheese
ADD

7:03

Brie

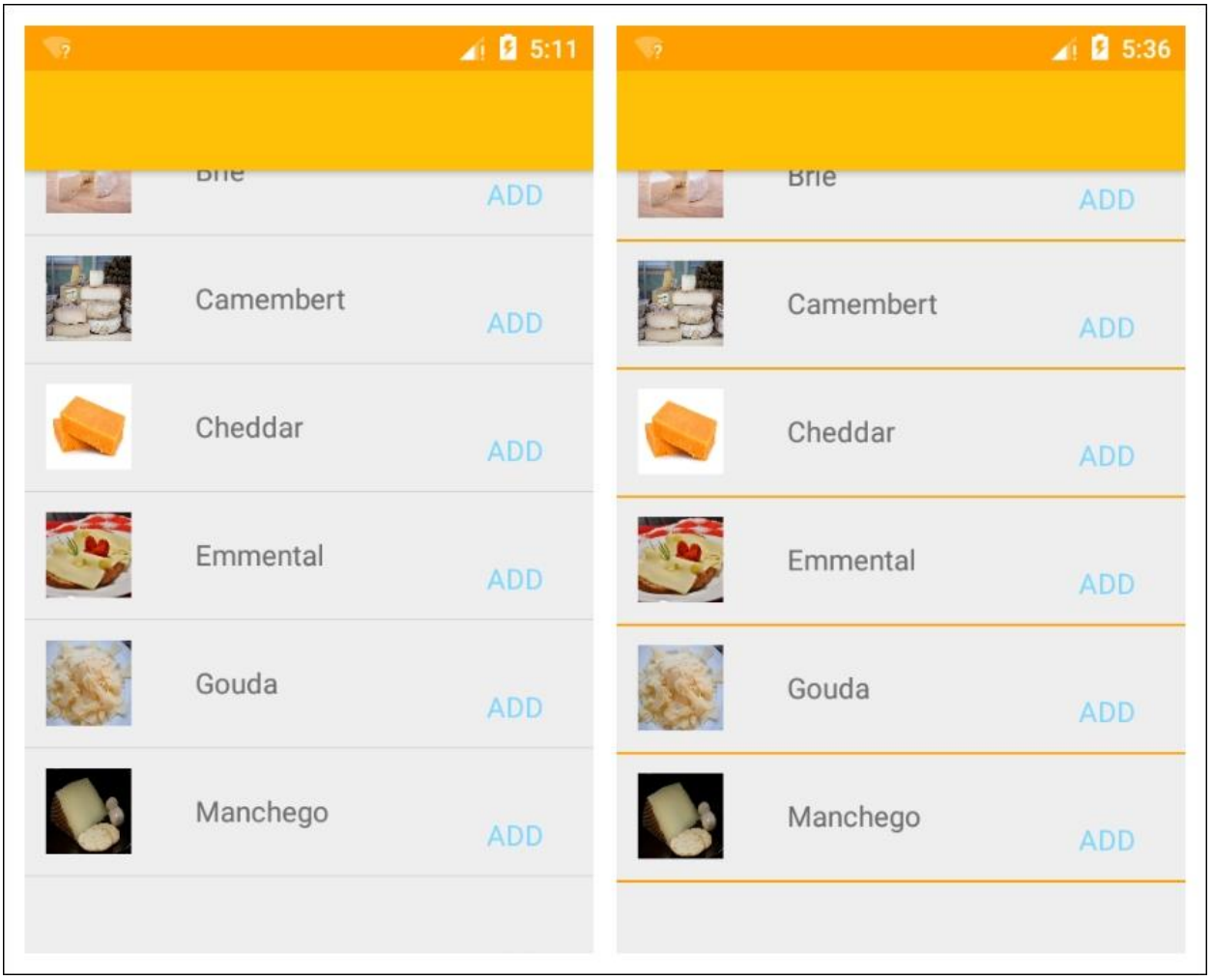
ADD

Camembert

ADD

Cheddar

ADD

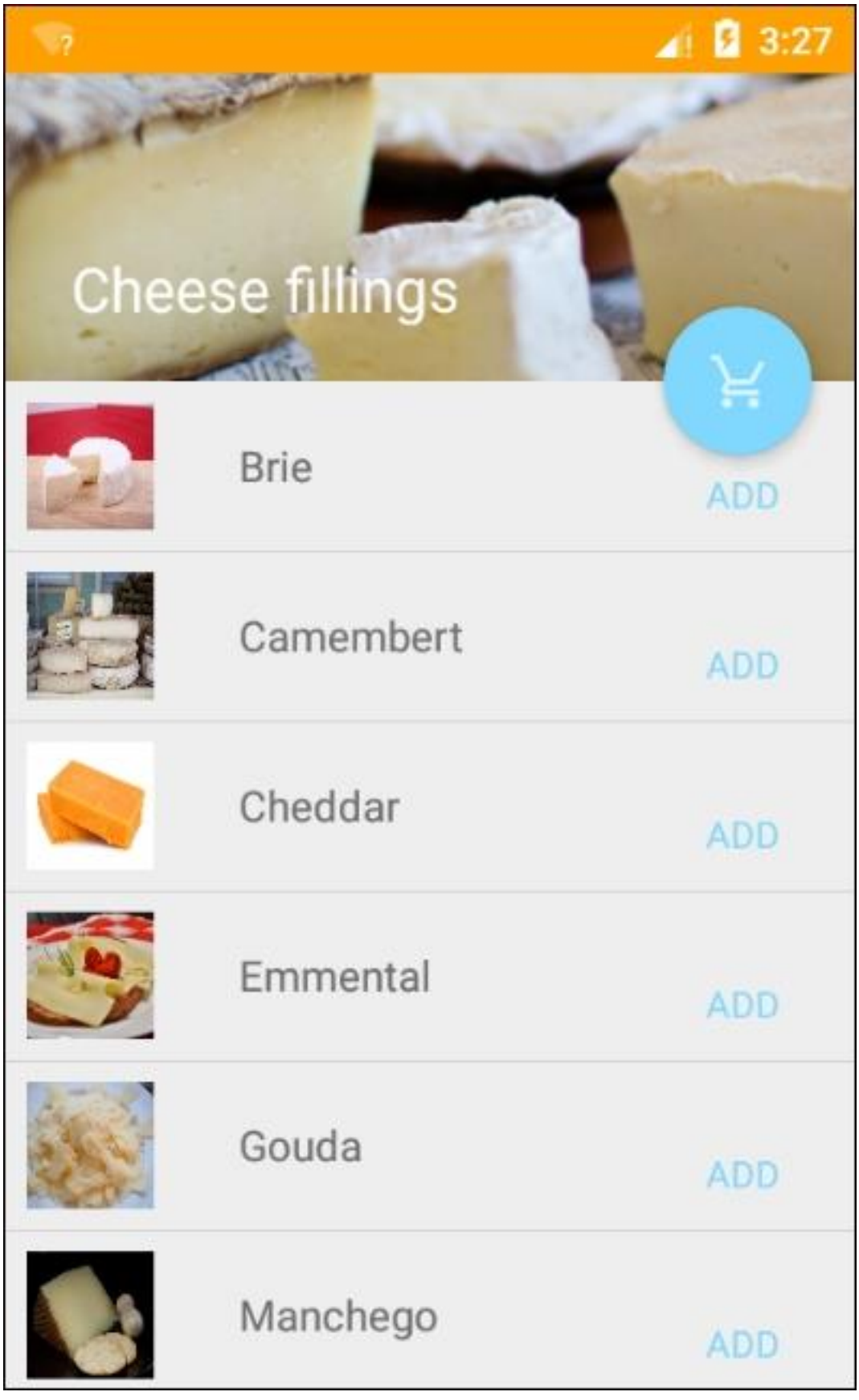

5:11
Brie
ADD
Camembert
ADD
Cheddar
ADD
Emmental
ADD
Gouda
ADD
Manchego
ADD
5:36
Brie
ADD
Camembert
ADD
Cheddar
ADD
Emmental
ADD
Gouda
ADD
Manchego
ADD

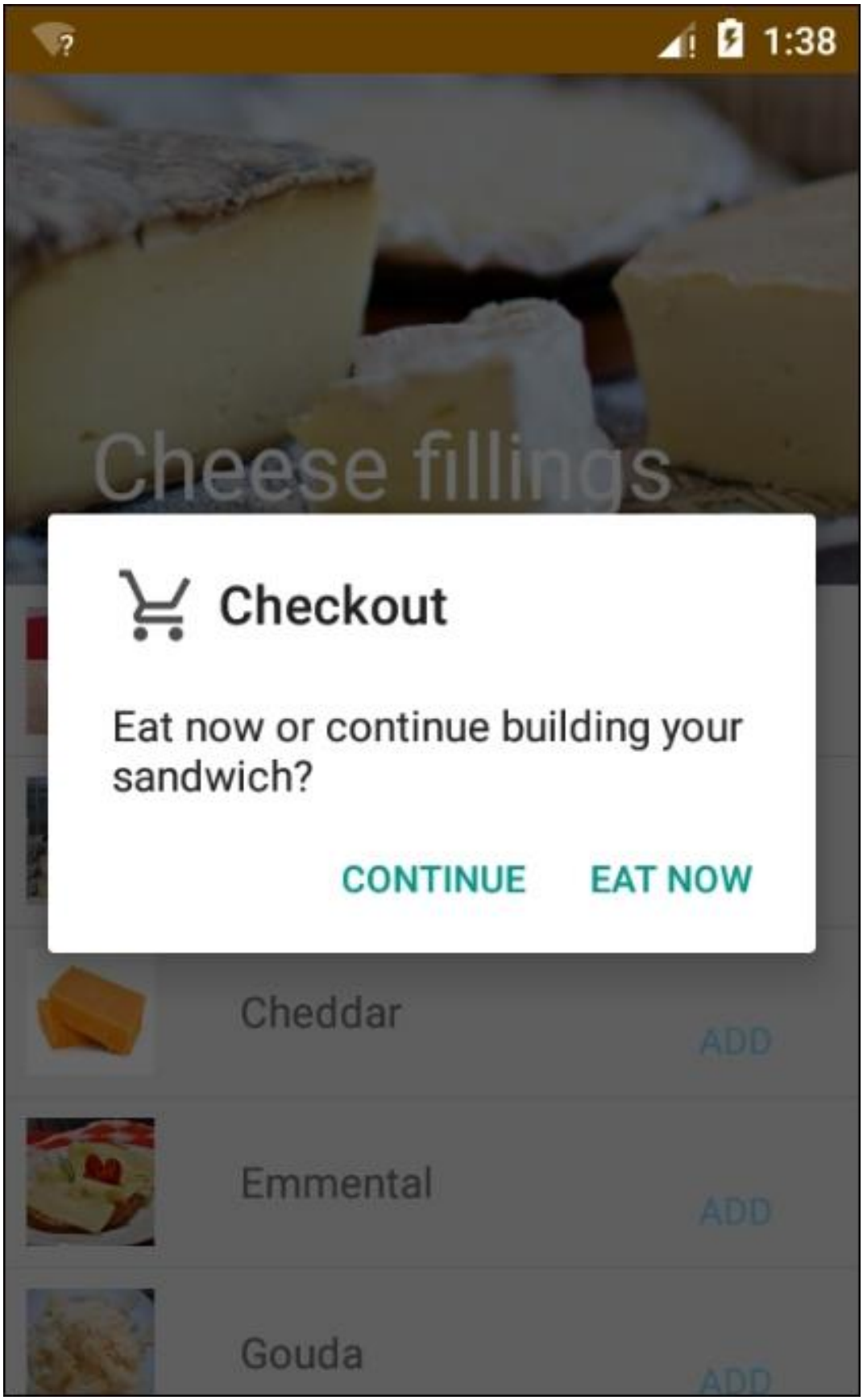
**Primary text** on a *dark background* is 87% opaque black: `#DE000000`

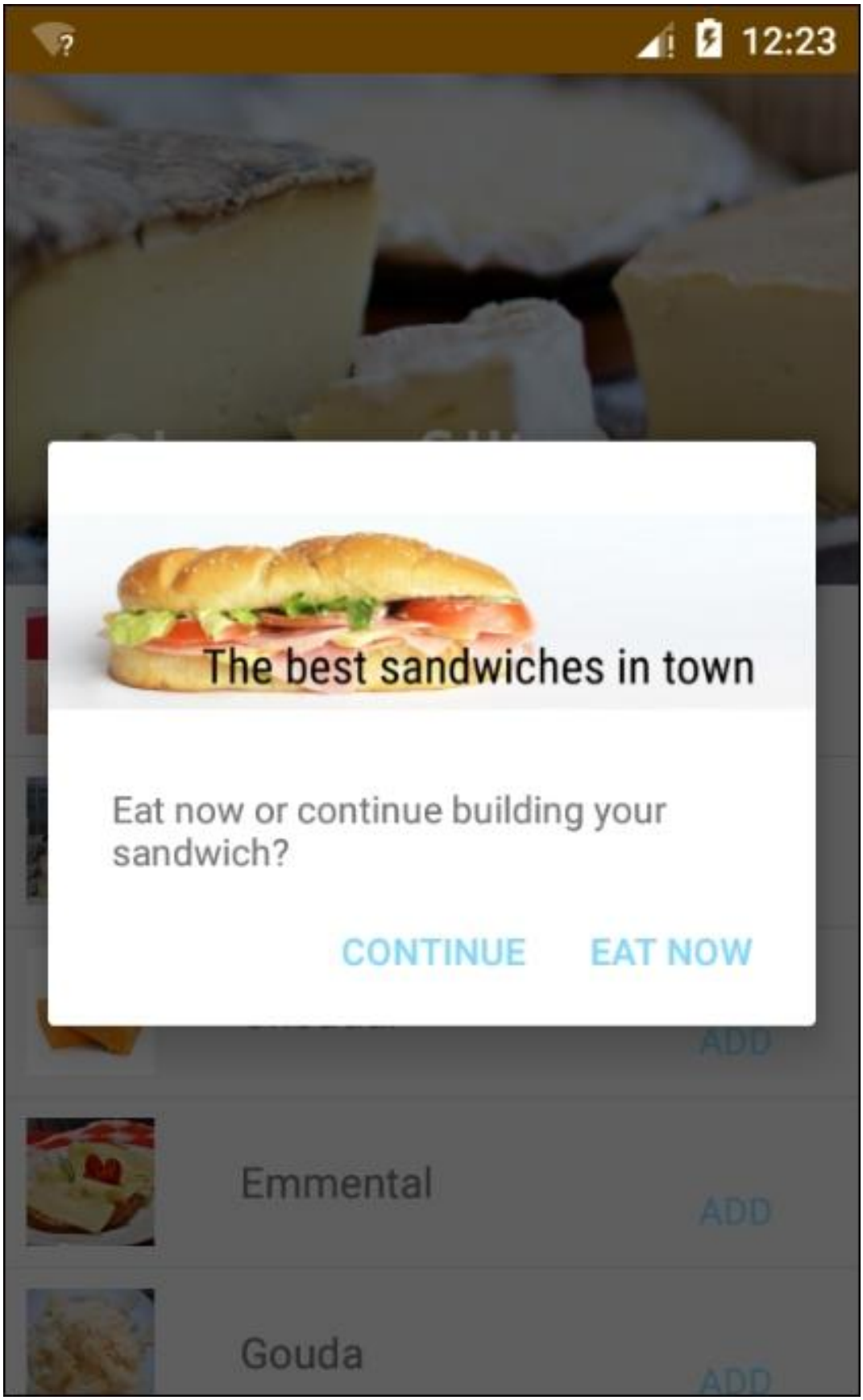
**Secondary text** on a *dark background* is 54% opaque black: `#8A000000`

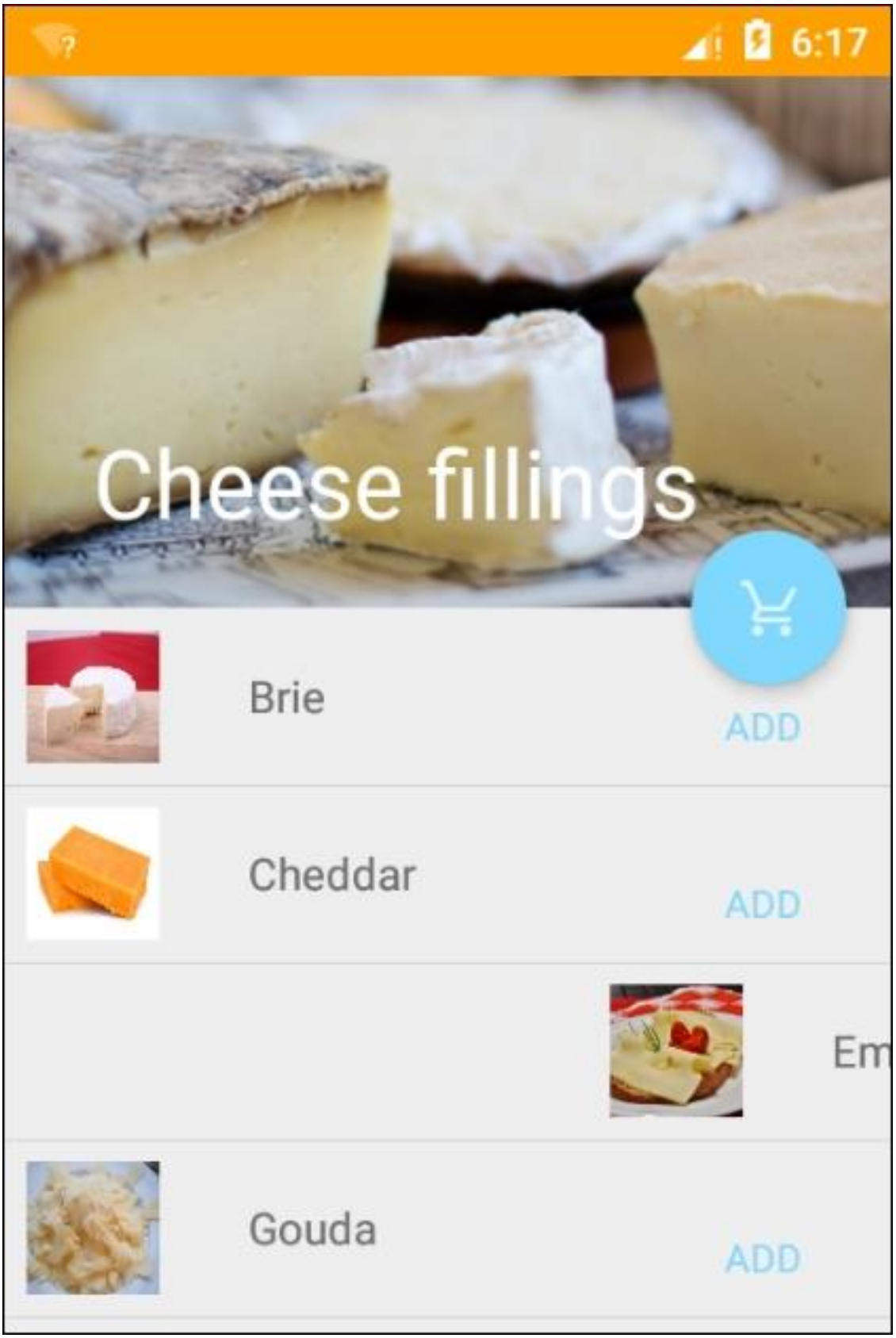
**Primary text** on a *light background* is 100% opaque white: `#FFFFFFFF`

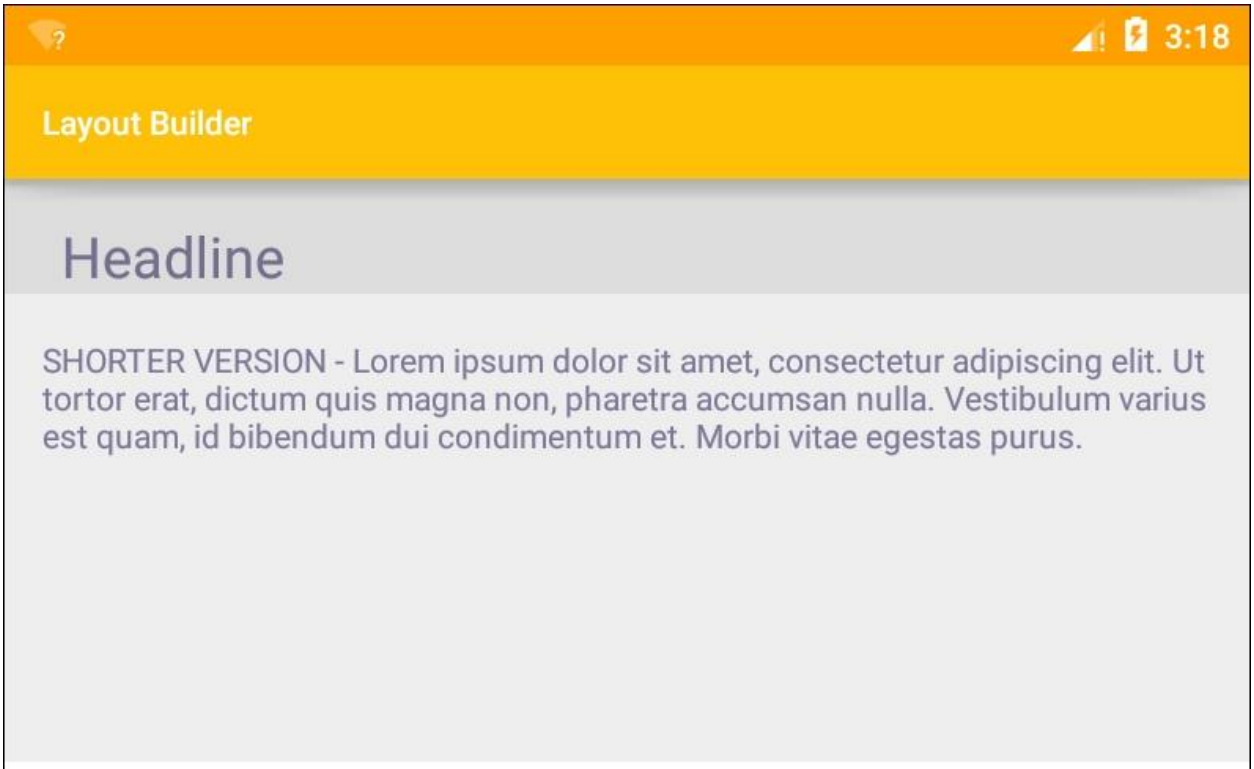
**Secondary text** on a *light background* is 70% opaque white: `#B3FFFFFF`

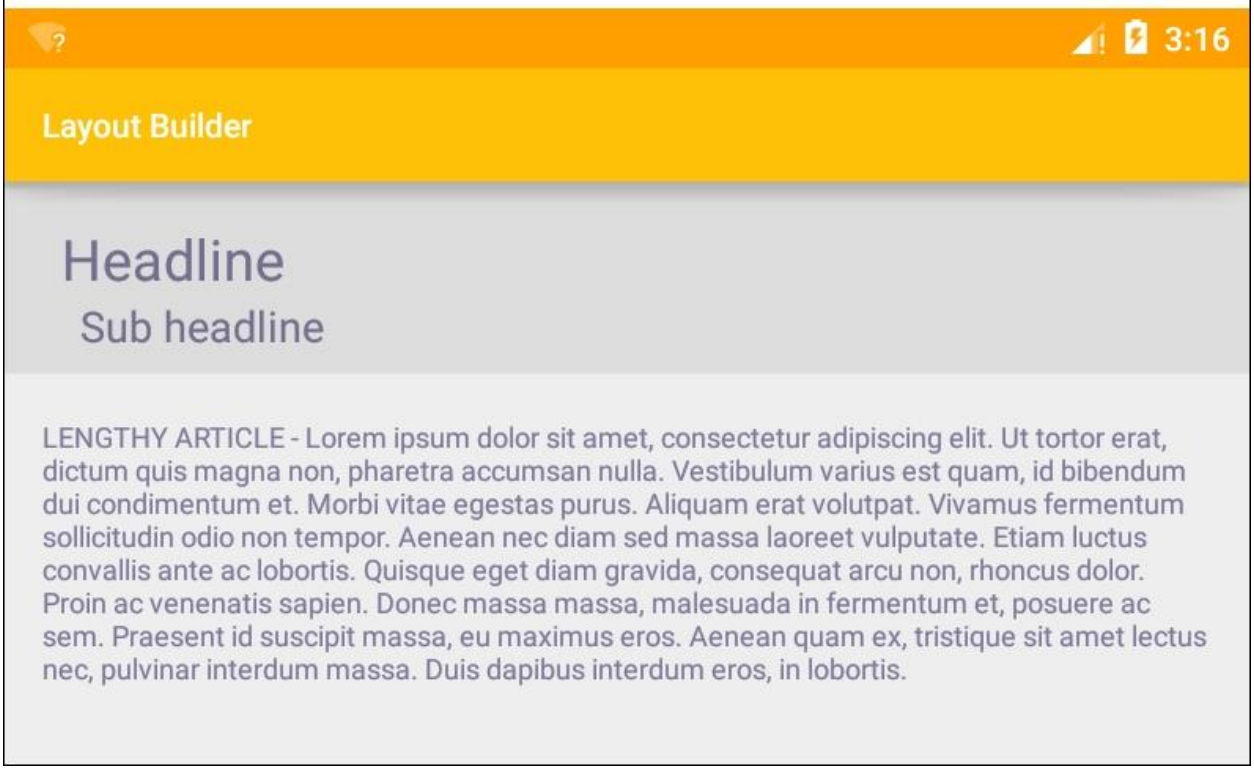

3:27
Cheese fillings
Brie
ADD
Camembert
ADD
Cheddar
ADD
Emmental
ADD
Gouda
ADD
Manchego
ADD

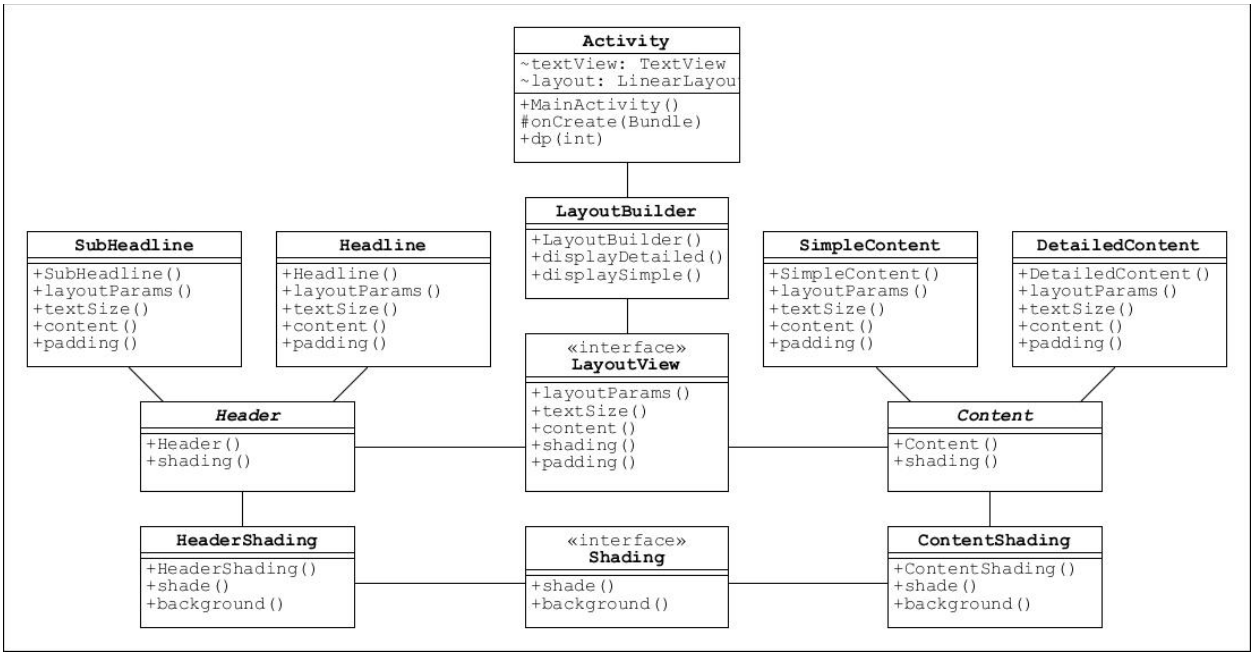

1:38
Cheese fillings
Checkout
Eat now or continue building your sandwich?
CONTINUE
EAT NOW
Cheddar
ADD
Emmental
ADD
Gouda
ADD


12:23
The best sandwiches in town
Eat now or continue building your sandwich?
CONTINUE
EAT NOW
Emmental
ADD
Gouda
ADD

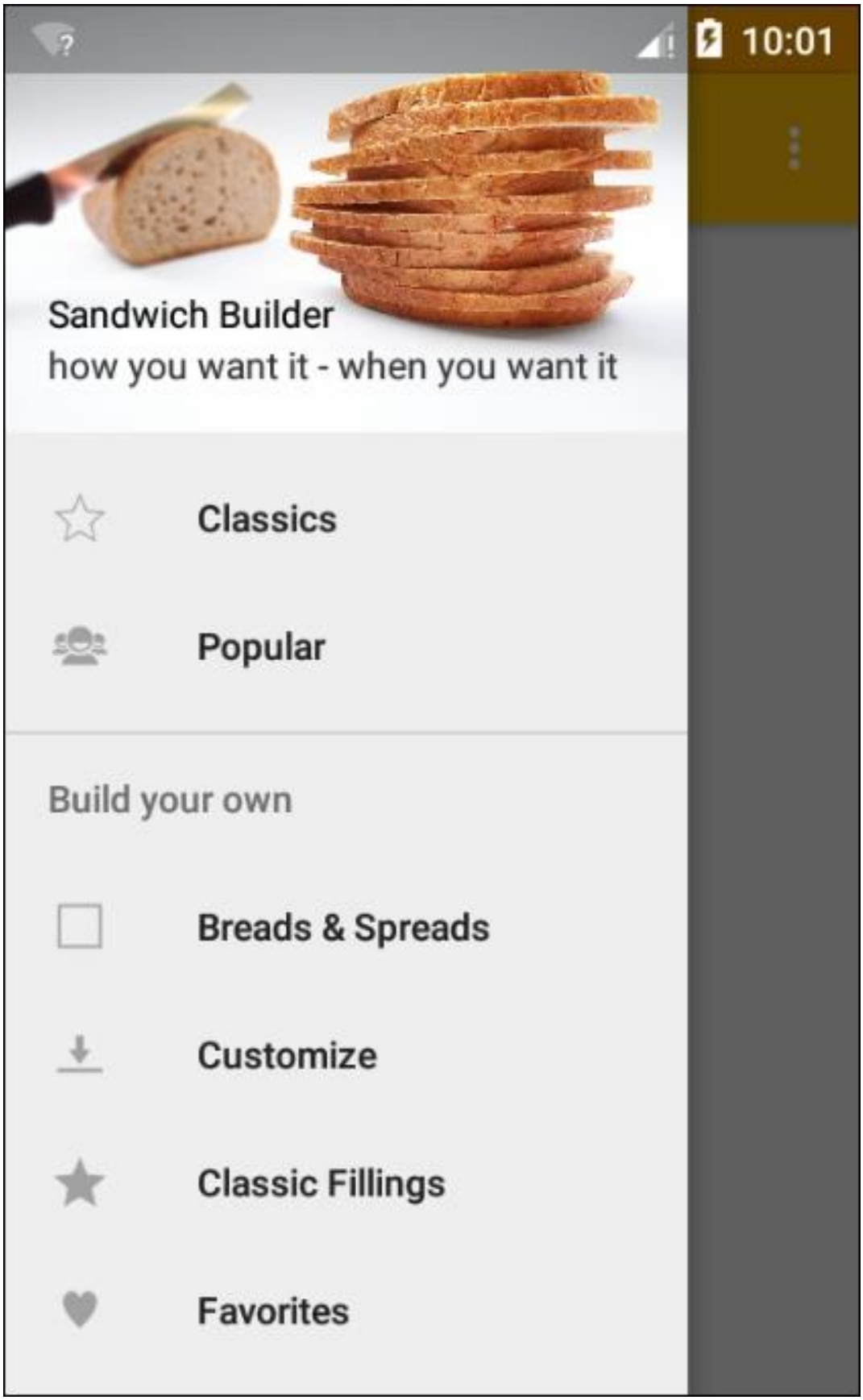

6:17
Cheese fillings
Brie
ADD
Cheddar
ADD
Em
Gouda
ADD

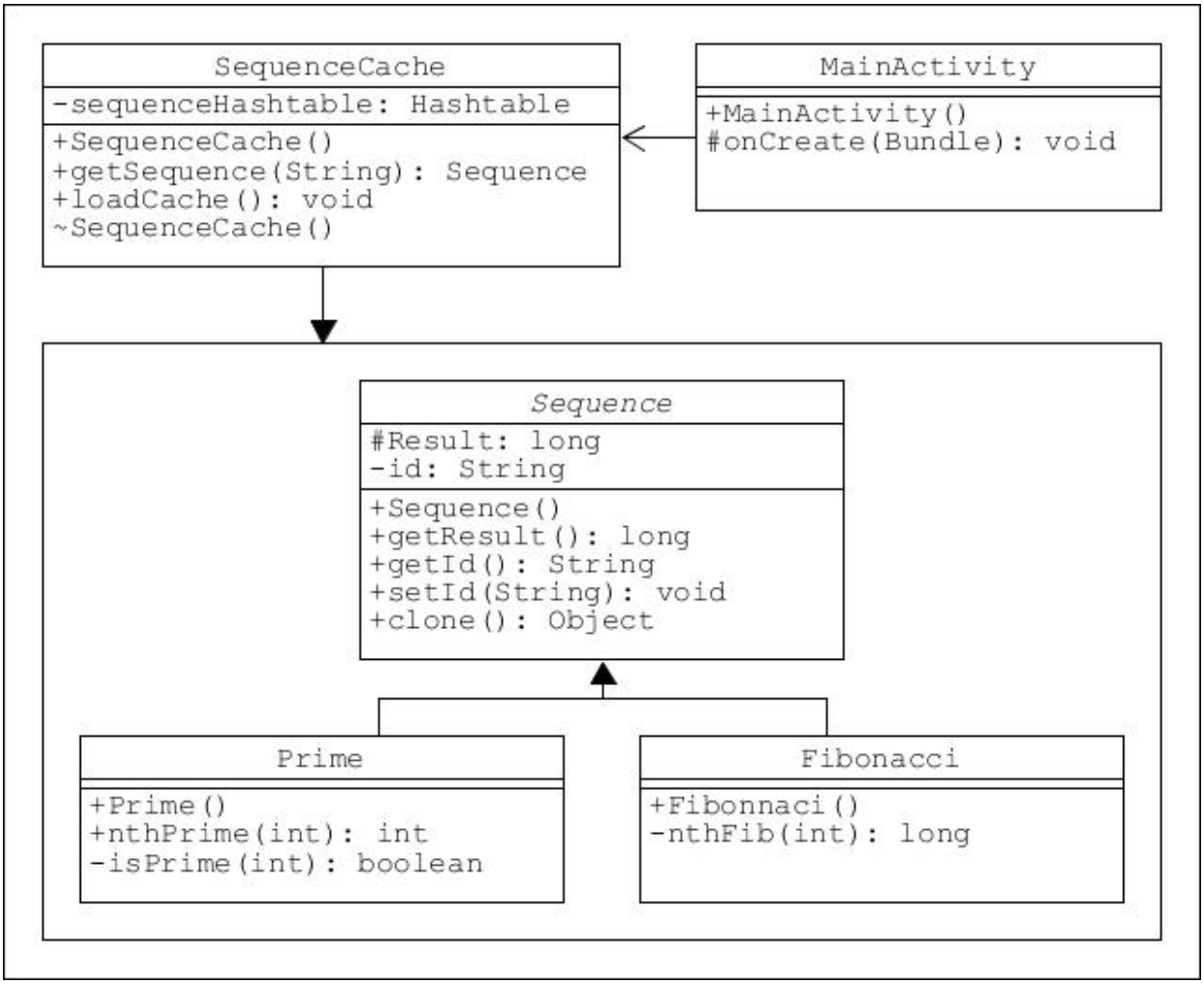

3:18
Layout Builder
Headline
SHORTER VERSION - Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut tortor erat, dictum quis magna non, pharetra accumsan nulla. Vestibulum varius est quam, id bibendum dui condimentum et. Morbi vitae egestas purus.

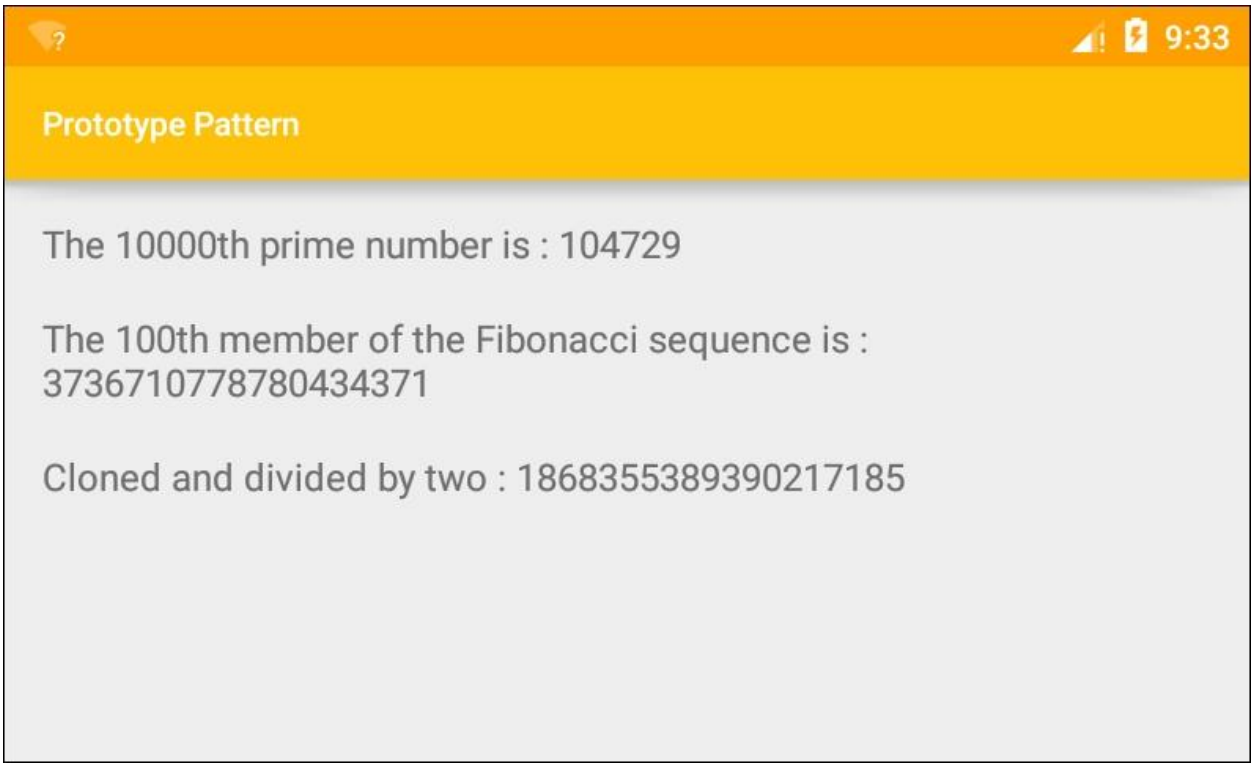

3:16
Layout Builder
Headline
Sub headline
LENGTHY ARTICLE - Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut tortor erat, dictum quis magna non, pharetra accumsan nulla. Vestibulum varius est quam, id bibendum dui condimentum et. Morbi vitae egestas purus. Aliquam erat volutpat. Vivamus fermentum sollicitudin odio non tempor. Aenean nec diam sed massa laoreet vulputate. Etiam luctus convallis ante ac lobortis. Quisque eget diam gravida, consequat arcu non, rhoncus dolor. Proin ac venenatis sapien. Donec massa massa, malesuada in fermentum et, posuere ac sem. Praesent id suscipit massa, eu maximus eros. Aenean quam ex, tristique sit amet lectus nec, pulvinar interdum massa. Duis dapibus interdum eros, in lobortis.

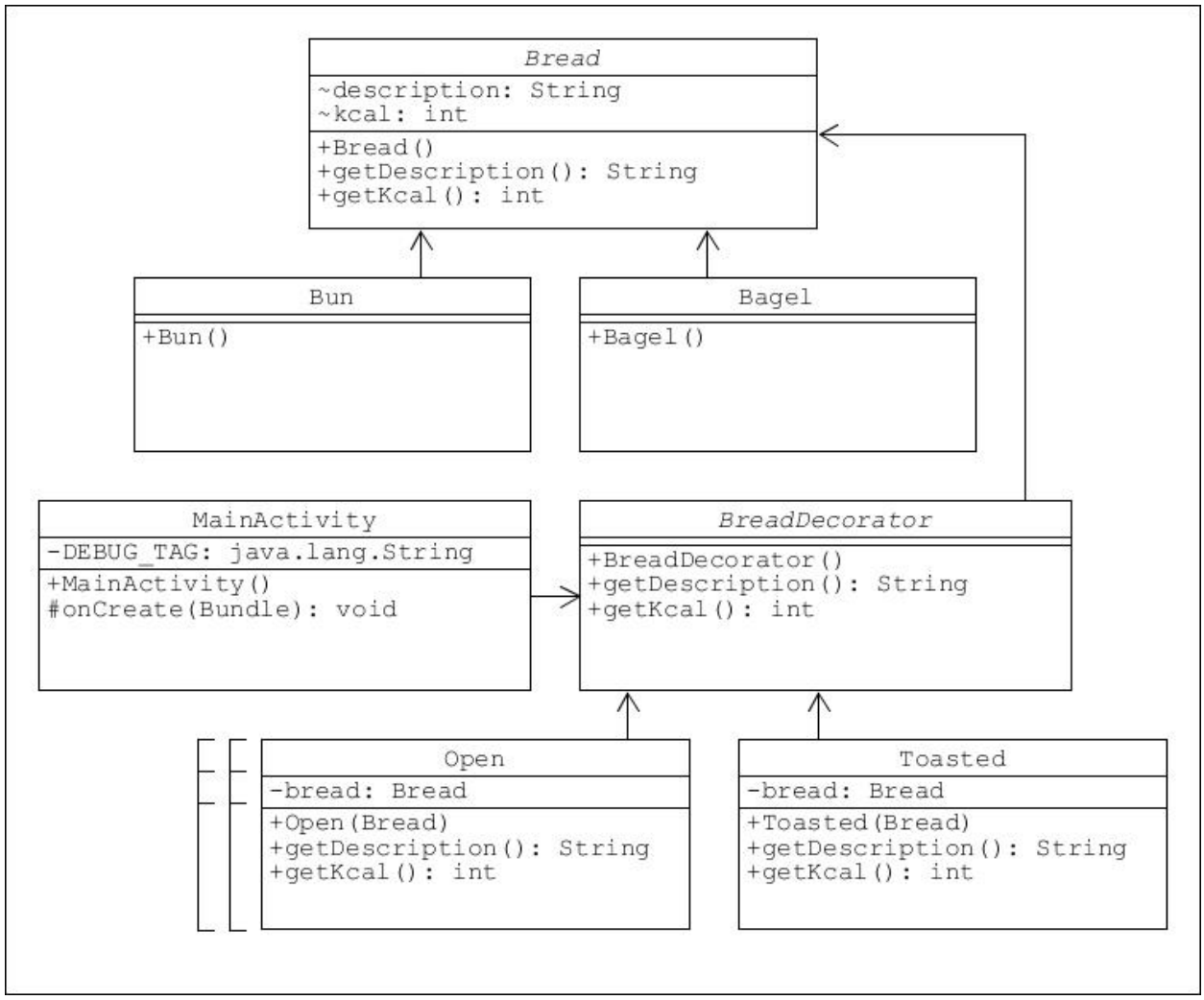

Activity
~textView: TextView
~layout: LinearLayou
+MainActivity()
#onCreate(Bundle)
+dp(int)
LayoutBuilder
+LayoutBuilder()
+displayDetailed()
+displaySimple()
SubHeadline
+SubHeadline()
+layoutParams()
+textSize()
+content()
+padding()
Headline
+Headline()
+layoutParams()
+textSize()
+content()
+padding()
SimpleContent
+SimpleContent()
+layoutParams()
+textSize()
+content()
+padding()
DetailedContent
+DetailedContent()
+layoutParams()
+textSize()
+content()
+padding()
«interface»
LayoutView
+layoutParams()
+textSize()
+content()
+shading()
+padding()
Header
+Header()
+shading()
Content
+Content()
+shading()
HeaderShading
+HeaderShading()
+shade()
+background()
«interface»
Shading
+shade()
+background()
ContentShading
+ContentShading()
+shade()
+background()

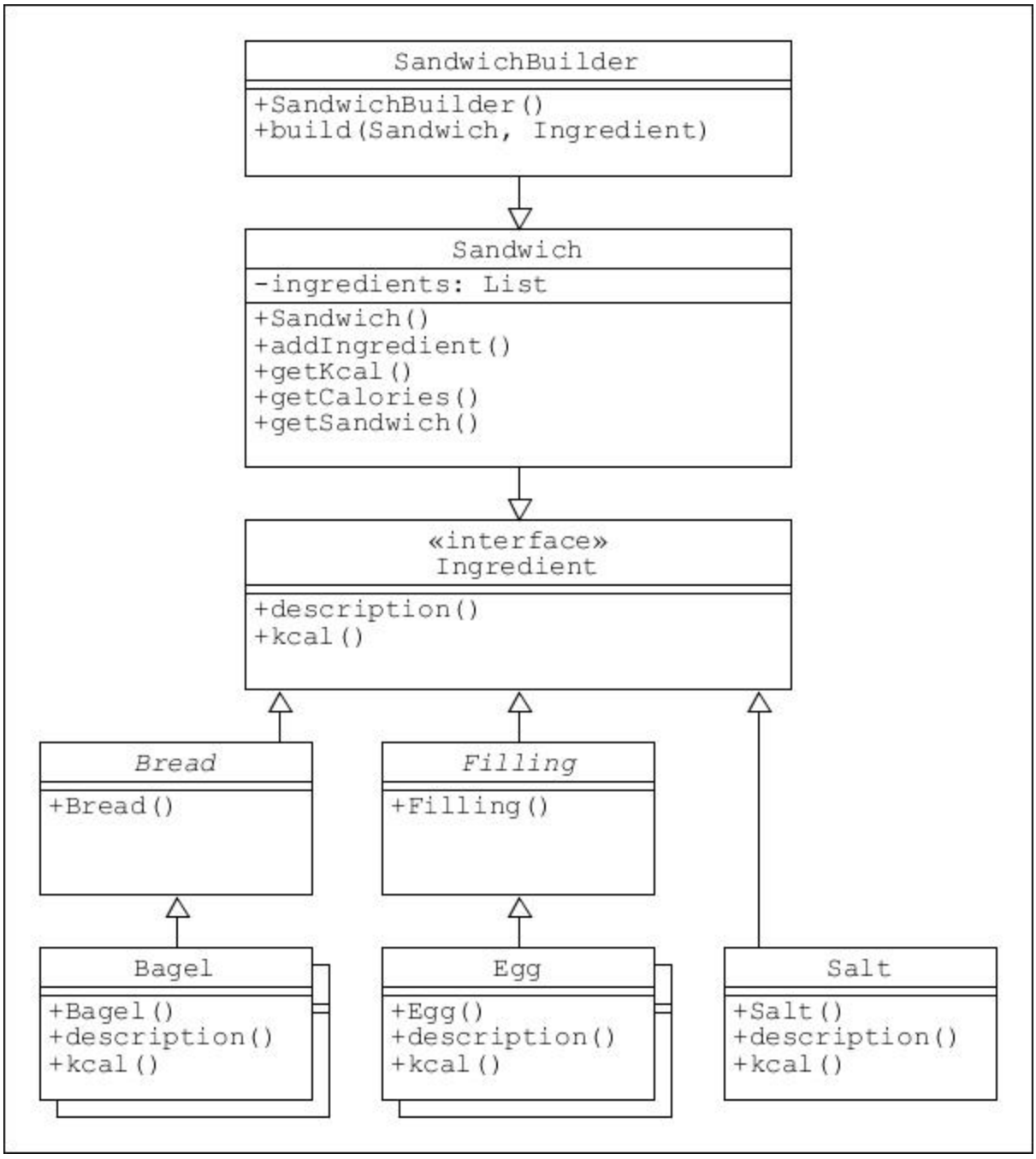
# Chapter 7: Combining Patterns

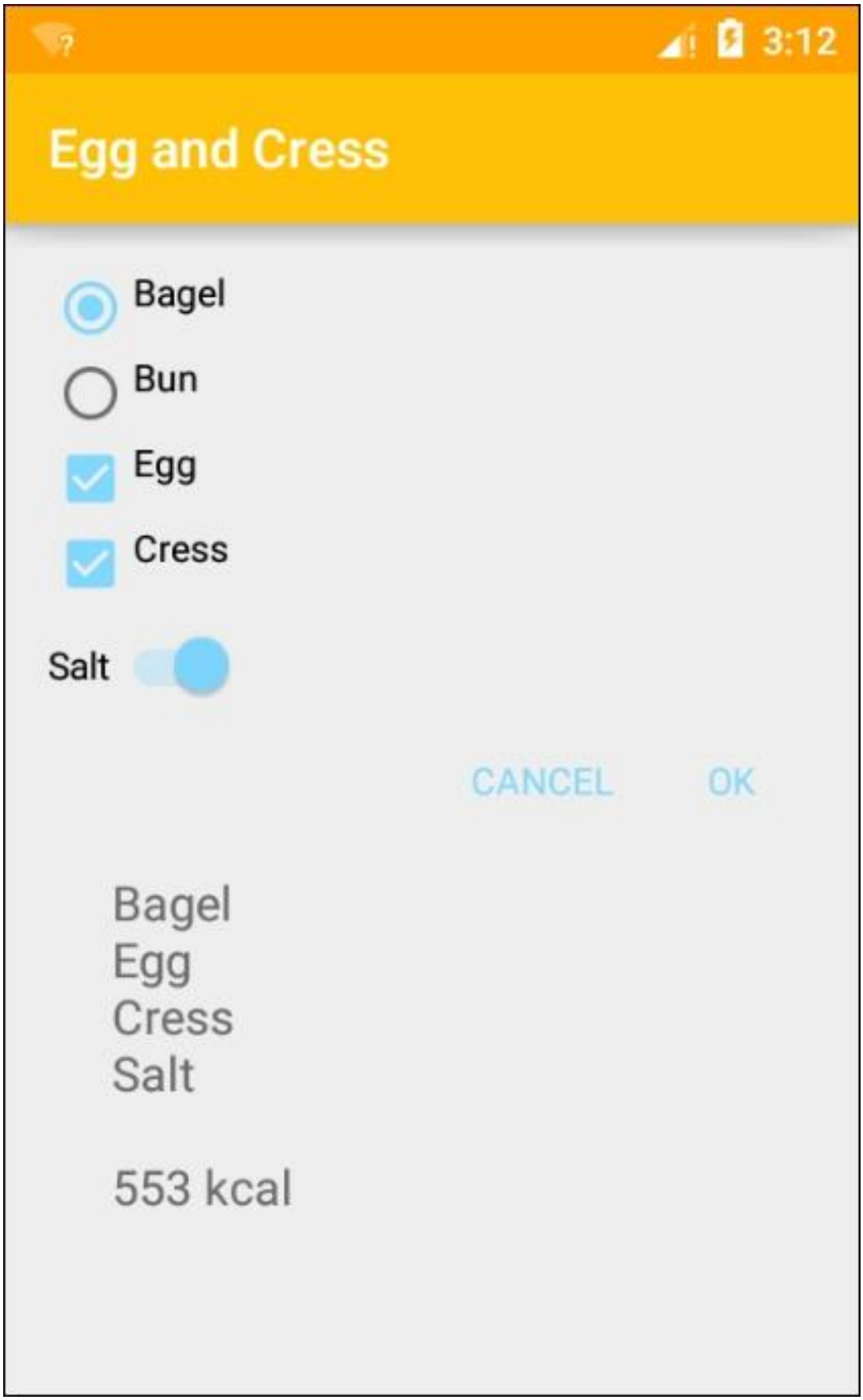

10:01
Sandwich Builder
how you want it - when you want it
Classics
Popular
Build your own
Breads & Spreads
Customize
Classic Fillings
Favorites

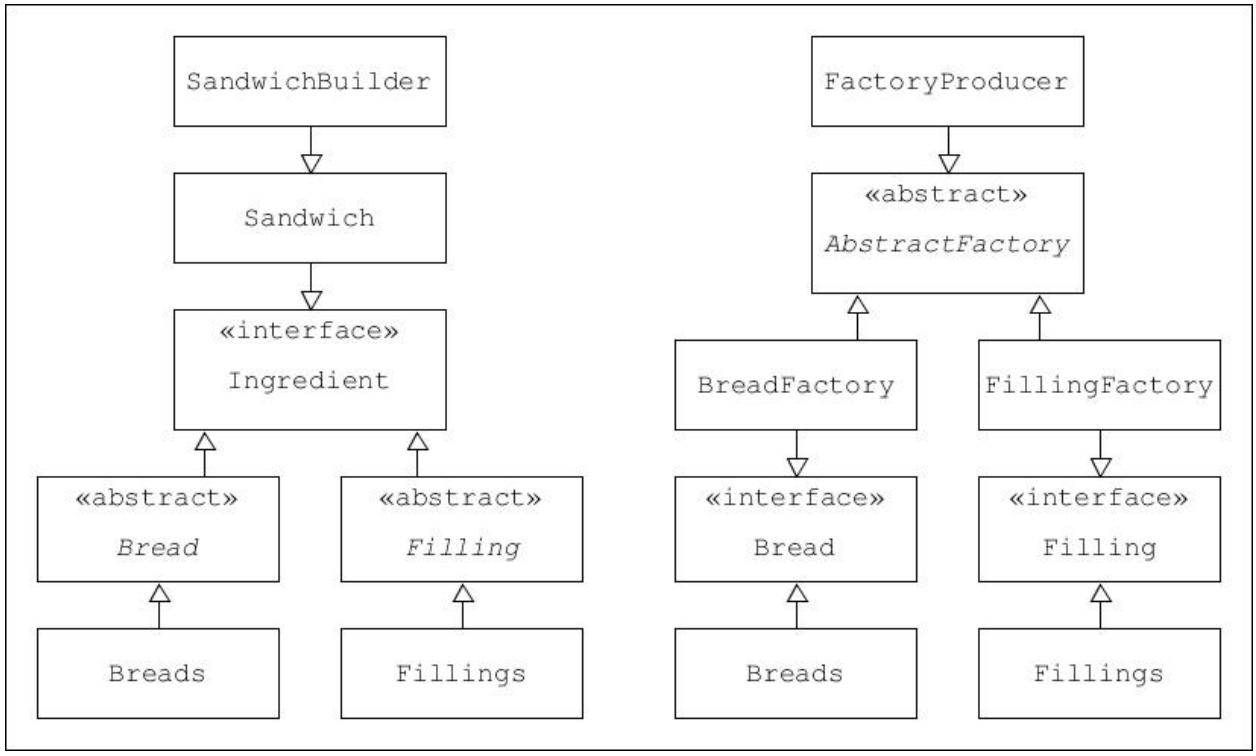

SequenceCache
-sequenceHashtable: Hashtable
+SequenceCache()
+getSequence(String): Sequence
+loadCache(): void
~SequenceCache()
MainActivity
+MainActivity()
#onCreate(Bundle): void
Sequence
#Result: long
-id: String
+Sequence()
+getResult(): long
+getId(): String
+setId(String): void
+clone(): Object
Prime
+Prime()
+nthPrime(int): int
-isPrime(int): boolean
Fibonacci
+Fibonnaci()
-nthFib(int): long

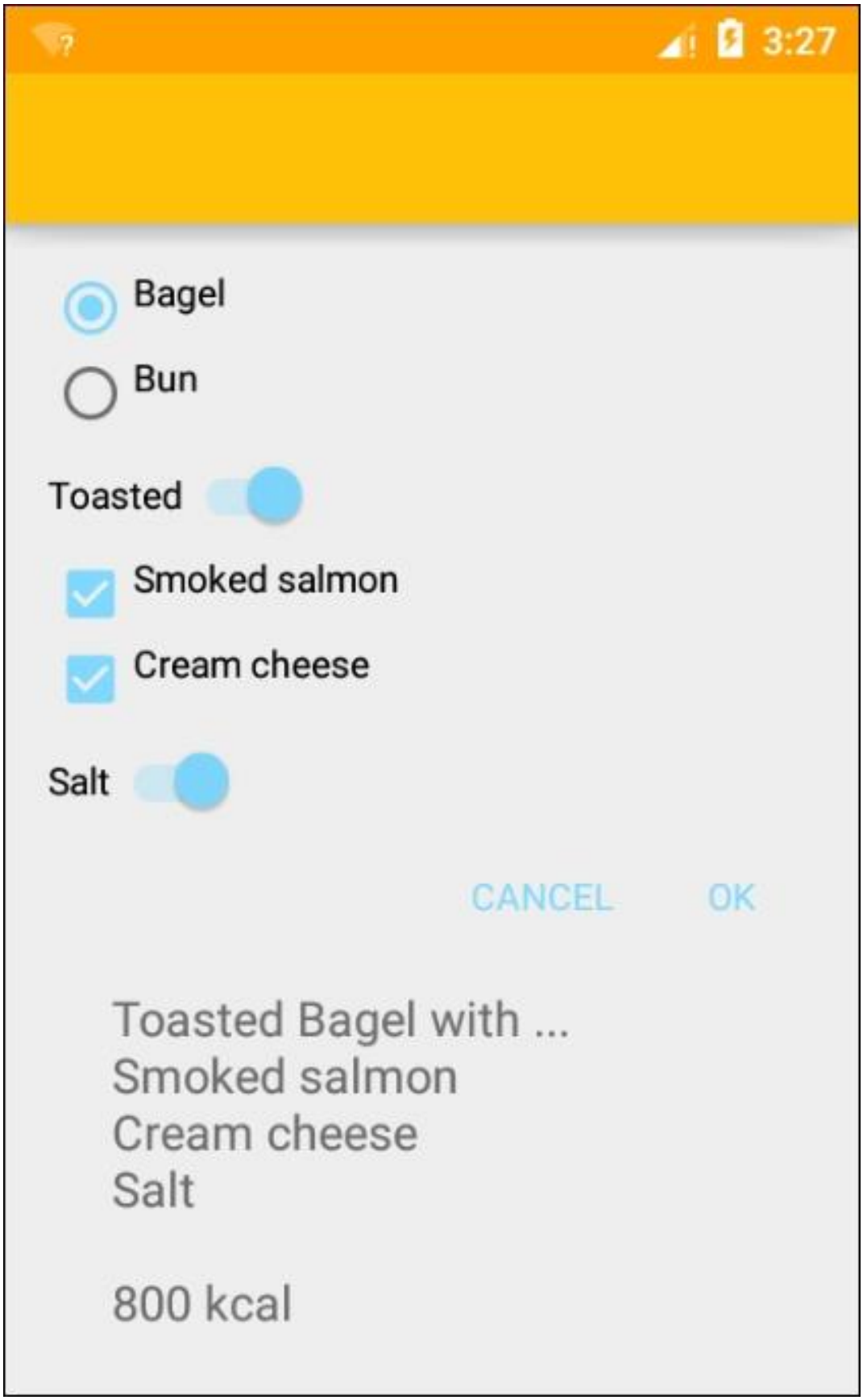

9:33
Prototype Pattern
The 10000th prime number is : 104729
The 100th member of the Fibonacci sequence is :
3736710778780434371
Cloned and divided by two : 1868355389390217185

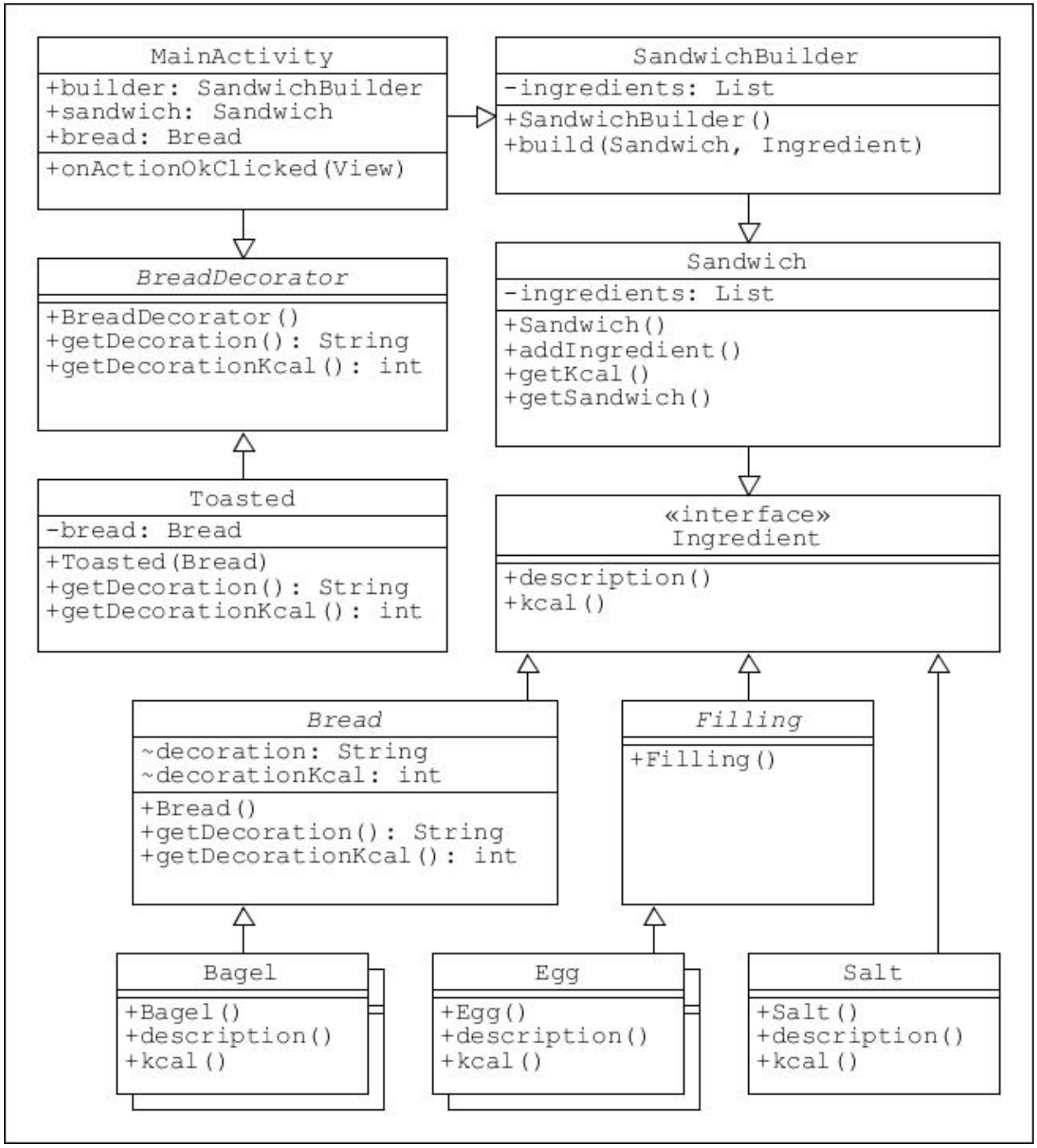

Bread
~description: String
~kcal: int
+Bread()
+getDescription(): String
+getKcal(): int
Bun
+Bun()
Bagel
+Bagel()
MainActivity
-DEBUG_TAG: java.lang.String
+MainActivity()
#onCreate(Bundle): void
BreadDecorator
+BreadDecorator()
+getDescription(): String
+getKcal(): int
Open
-bread: Bread
+Open(Bread)
+getDescription(): String
+getKcal(): int
Toasted
-bread: Bread
+Toasted(Bread)
+getDescription(): String
+getKcal(): int

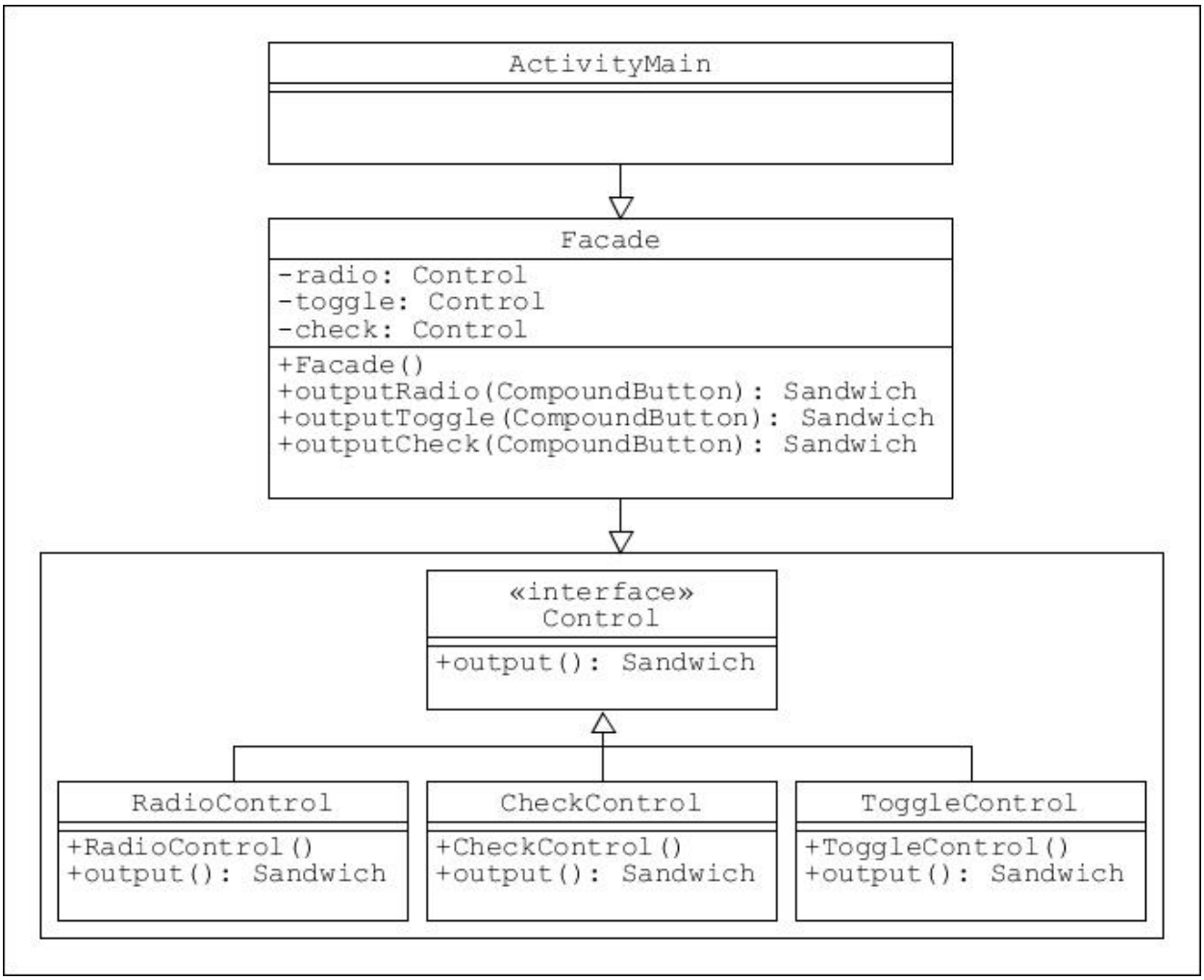

SandwichBuilder
+SandwichBuilder()
+build(Sandwich, Ingredient)
Sandwich
-ingredients: List
+Sandwich()
+addIngredient()
+getKcal()
+getCalories()
+getSandwich()
«interface»
Ingredient
+description()
+kcal()
Bread
+Bread()
Filling
+Filling()
Bagel
+Bagel()
+description()
+kcal()
Egg
+Egg()
+description()
+kcal()
Salt
+Salt()
+description()
+kcal()


3:12
Egg and Cress
Bagel
Bun
Egg
Cress
Salt
CANCEL
OK
Bagel
Egg
Cress
Salt
553 kcal

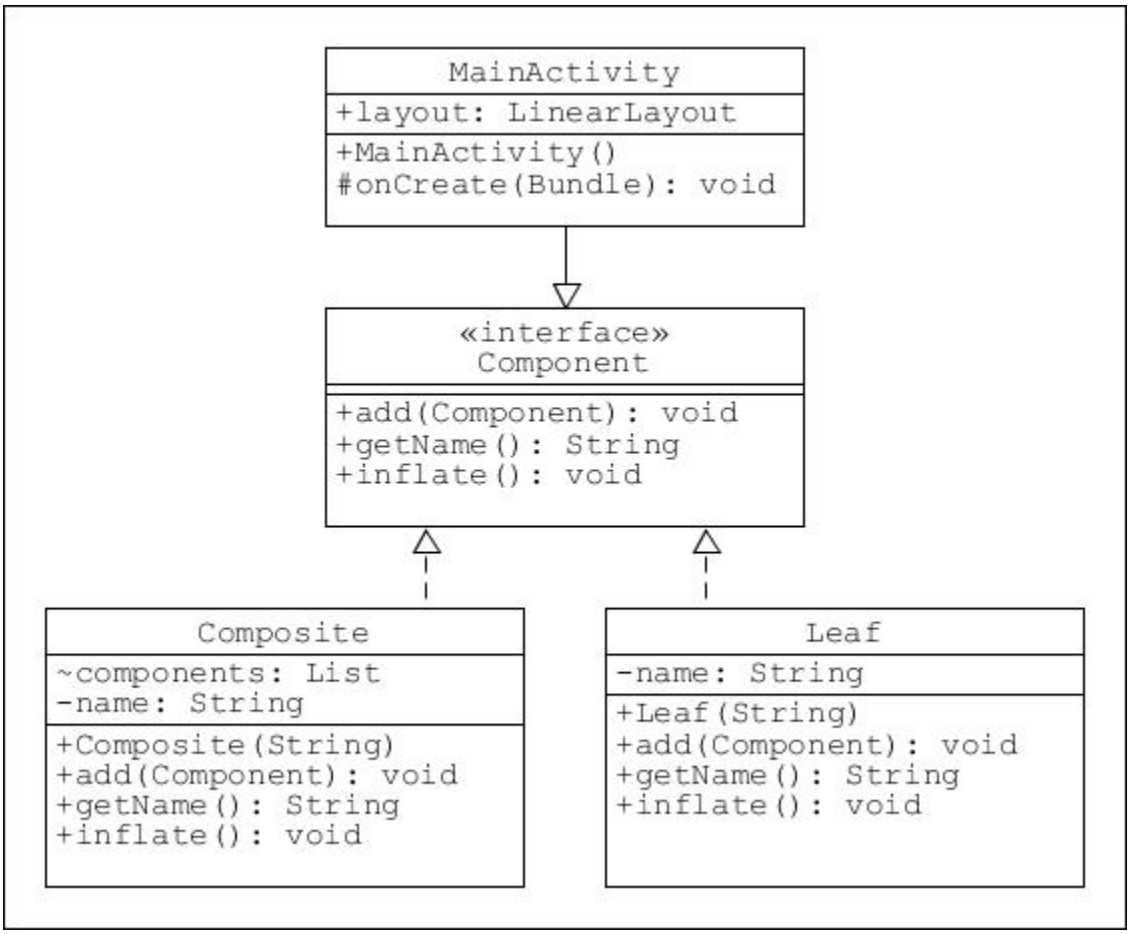

SandwichBuilder
Sandwich
«interface»
Ingredient
«abstract»
Bread
«abstract»
Filling
Breads
Fillings
FactoryProducer
«abstract»
AbstractFactory
BreadFactory
FillingFactory
«interface»
Bread
«interface»
Filling
Breads
Fillings

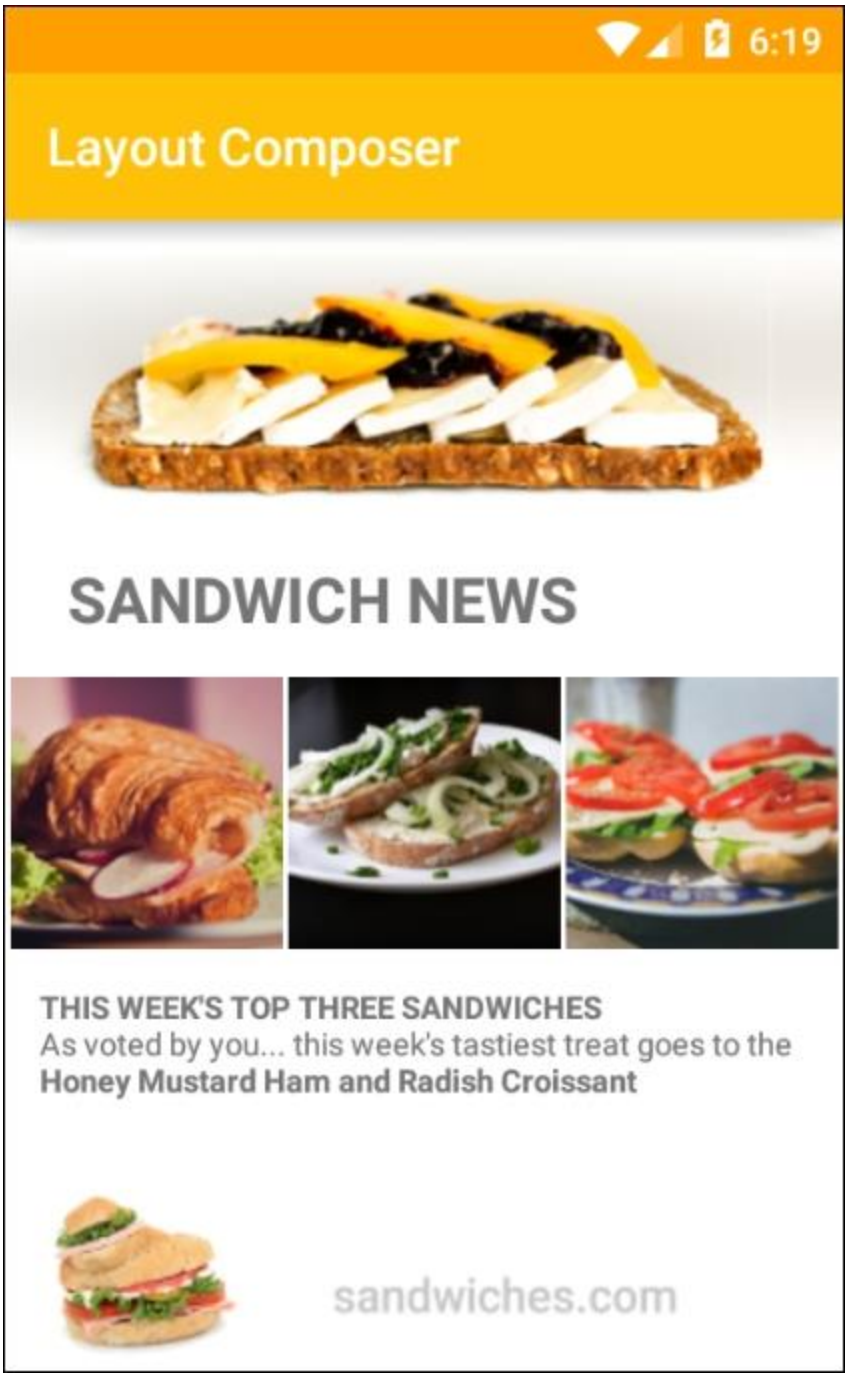

3:27
Bagel
Bun
Toasted
Smoked salmon
Cream cheese
Salt
CANCEL
OK
Toasted Bagel with ...
Smoked salmon
Cream cheese
Salt
800 kcal

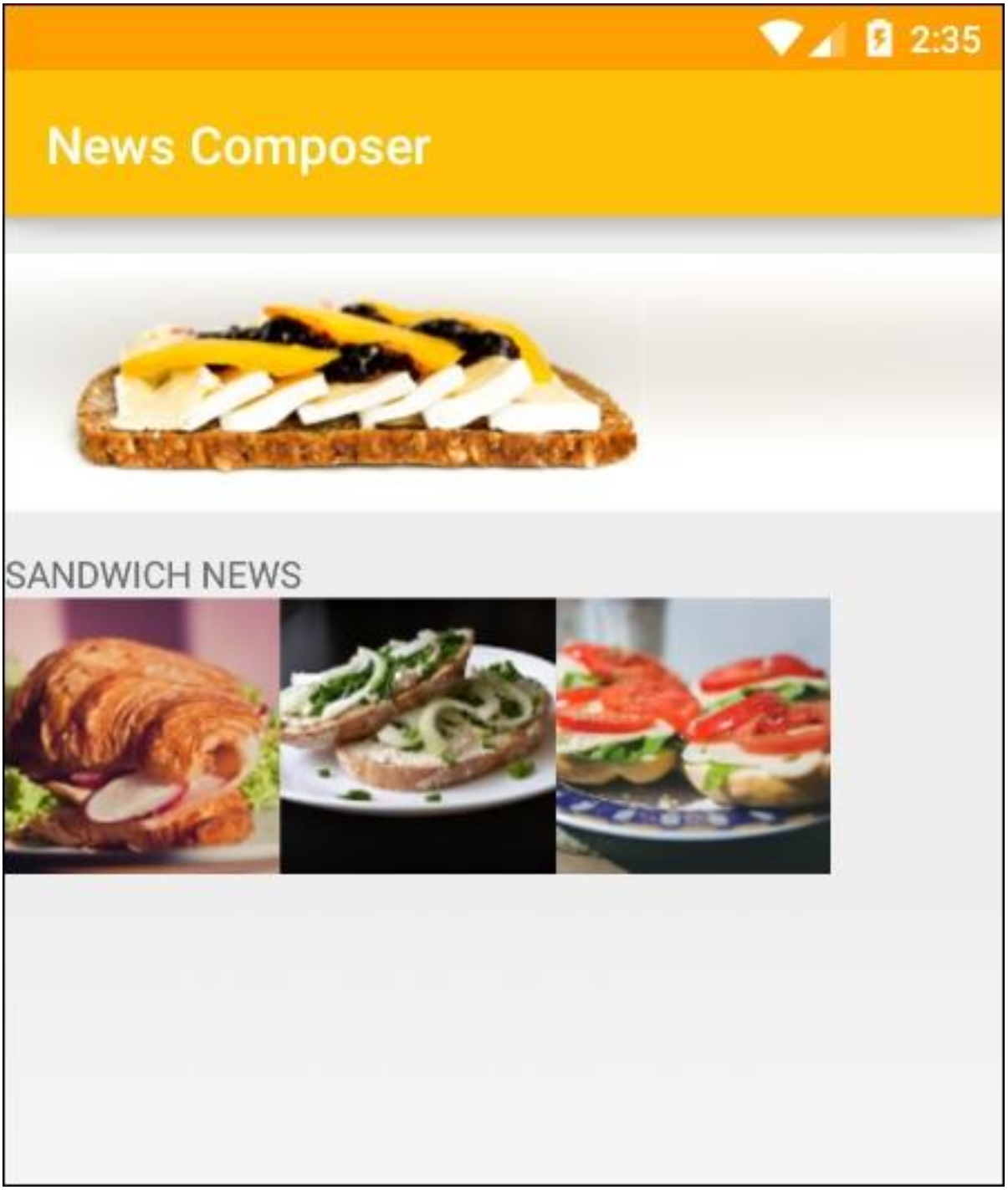

MainActivity
+builder: SandwichBuilder
+sandwich: Sandwich
+bread: Bread
+onActionOkClicked(View)
SandwichBuilder
-ingredients: List
+SandwichBuilder()
+build(Sandwich, Ingredient)
BreadDecorator
+BreadDecorator()
+getDecoration(): String
+getDecorationKcal(): int
Sandwich
-ingredients: List
+Sandwich()
+addIngredient()
+getKcal()
+getSandwich()
Toasted
-bread: Bread
+Toasted(Bread)
+getDecoration(): String
+getDecorationKcal(): int
«interface»
Ingredient
+description()
+kcal()
Bread
~decoration: String
~decorationKcal: int
+Bread()
+getDecoration(): String
+getDecorationKcal(): int
Filling
+Filling()
Bagel
+Bagel()
+description()
+kcal()
Egg
+Egg()
+description()
+kcal()
Salt
+Salt()
+description()
+kcal()

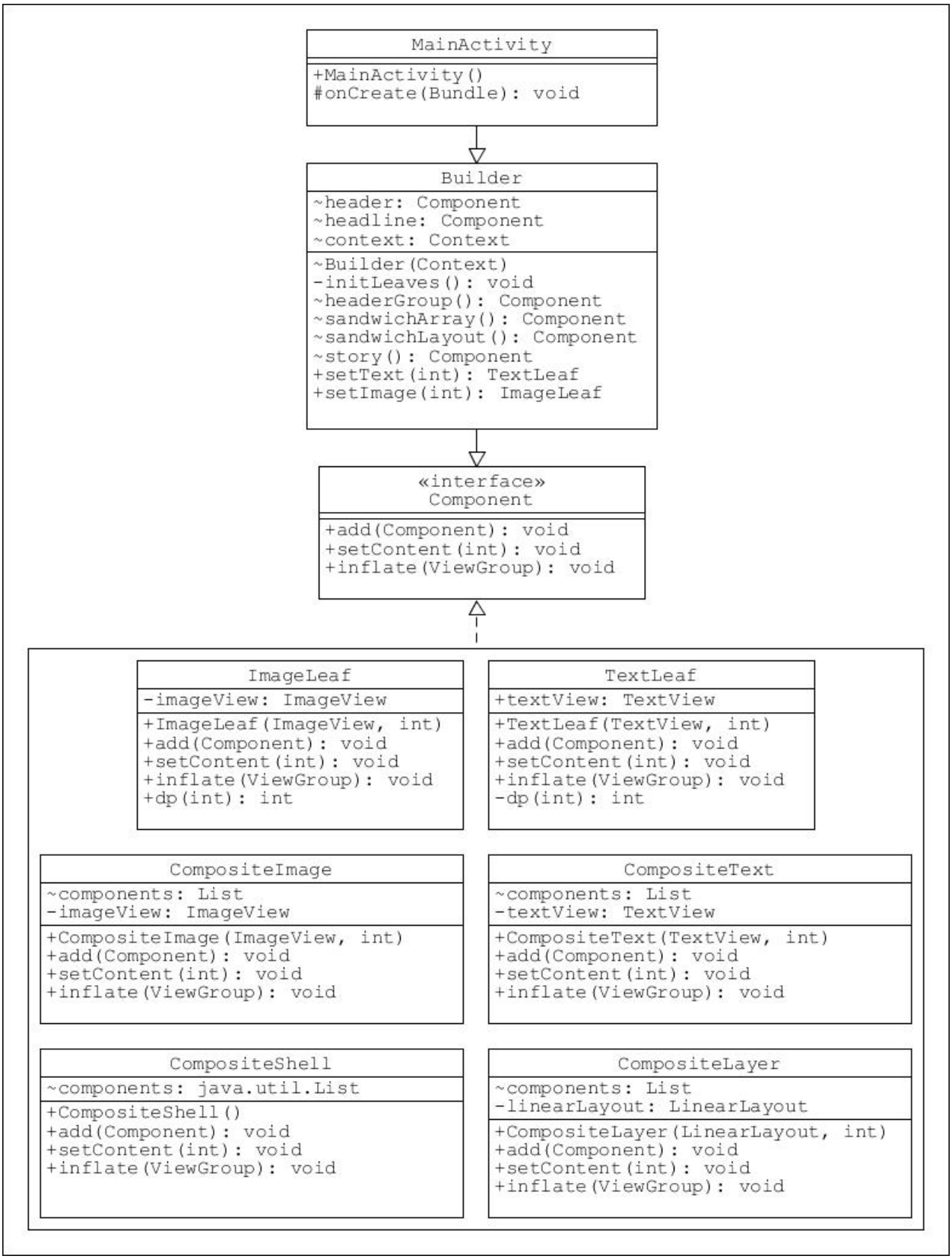

ActivityMain
Facade
-radio: Control
-toggle: Control
-check: Control
+Facade()
+outputRadio(CompoundButton): Sandwich
+outputToggle(CompoundButton): Sandwich
+outputCheck(CompoundButton): Sandwich
«interface»
Control
+output(): Sandwich
RadioControl
+RadioControl()
+output(): Sandwich
CheckControl
+CheckControl()
+output(): Sandwich
ToggleControl
+ToggleControl()
+output(): Sandwich

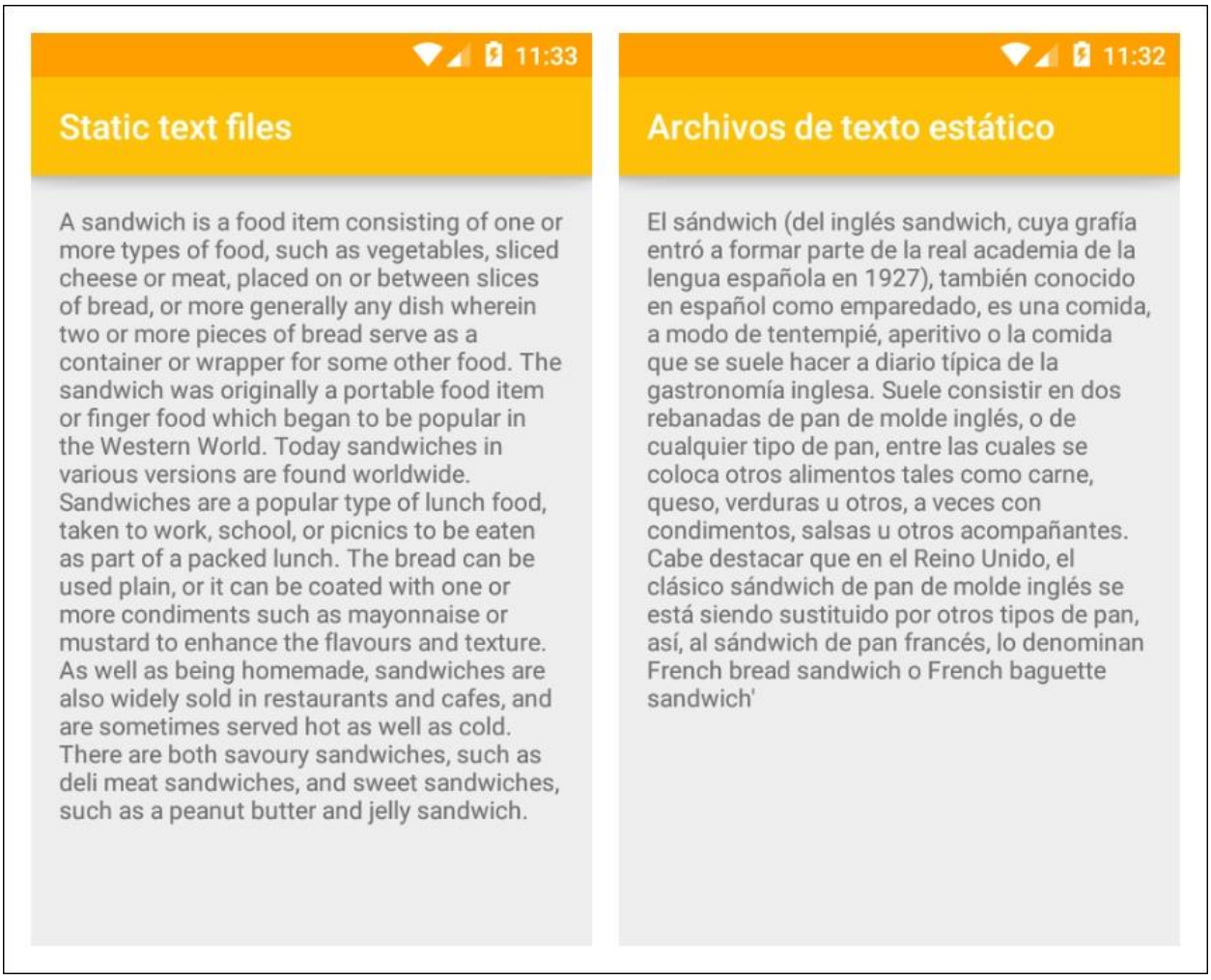
## Chapter 8: Composing Patterns

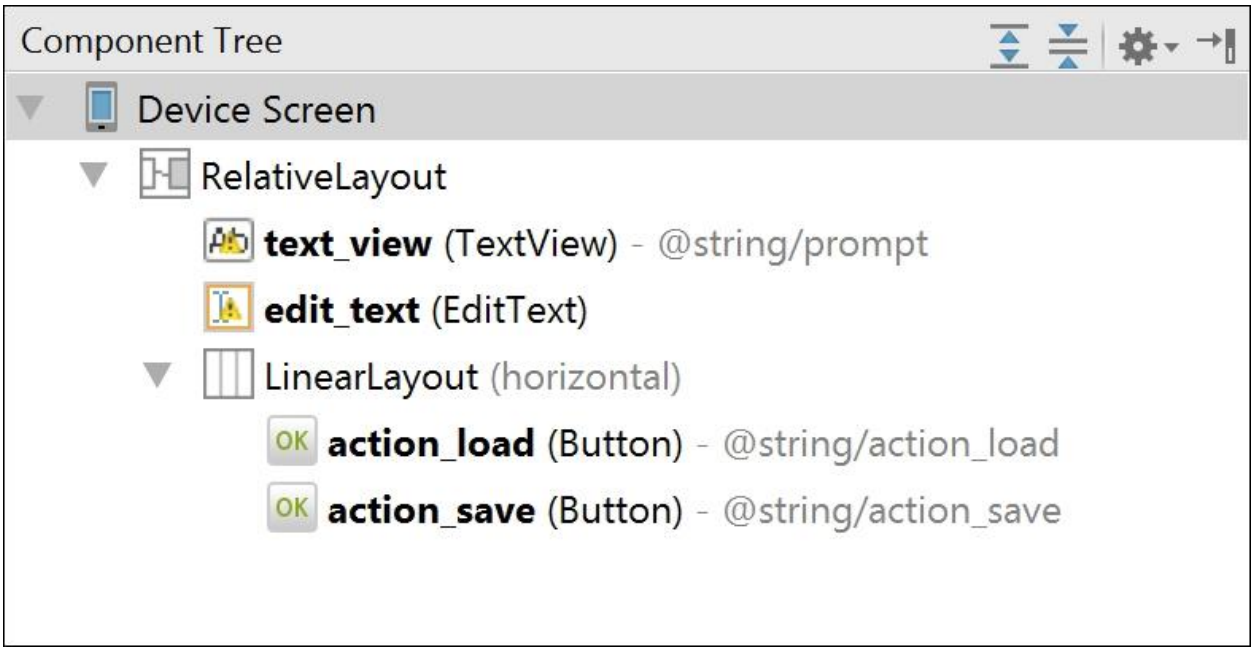


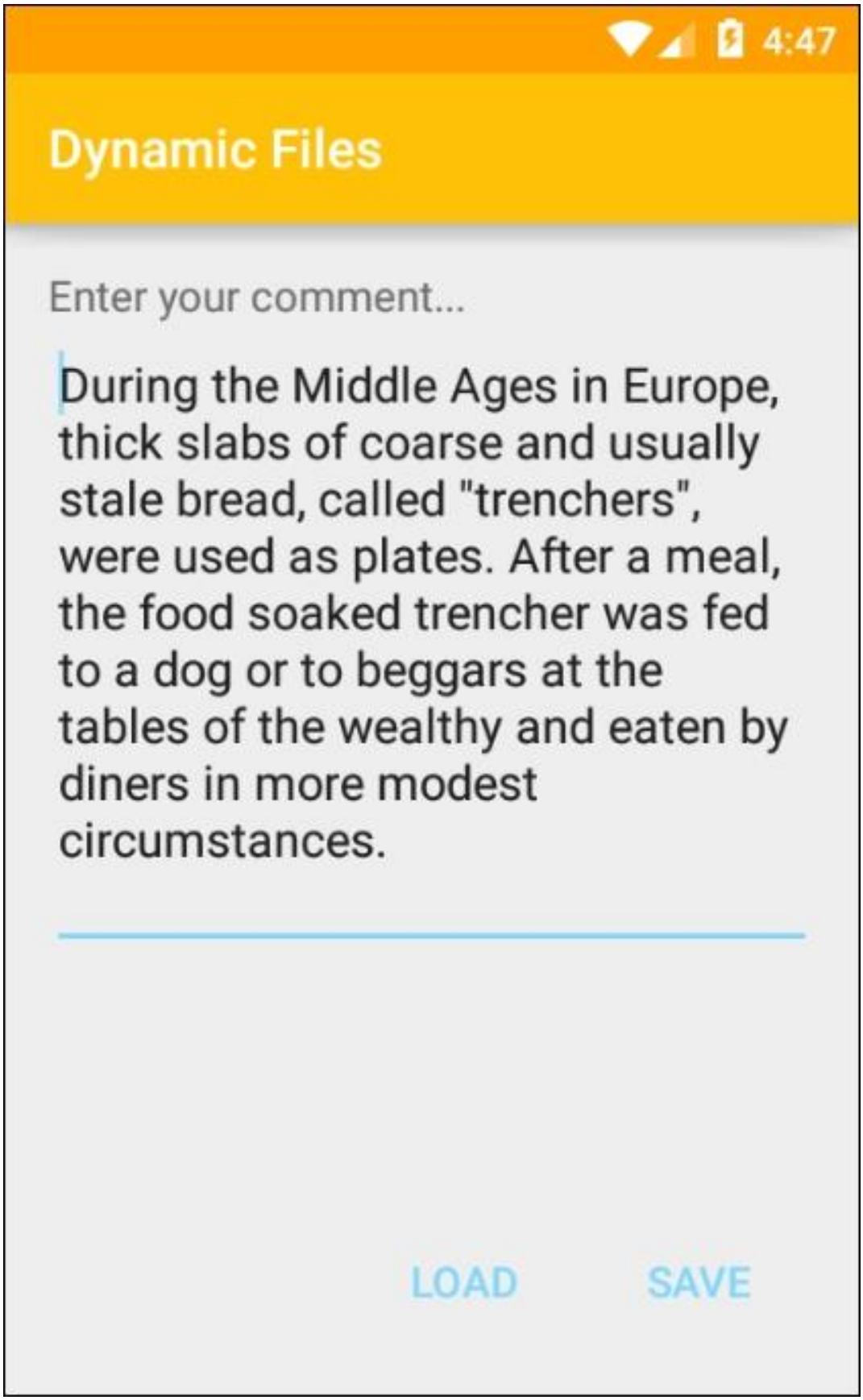

6:19
Layout Composer
SANDWICH NEWS
THIS WEEK'S TOP THREE SANDWICHES
As voted by you... this week's tastiest treat goes to the
Honey Mustard Ham and Radish Croissant
sandwiches.com

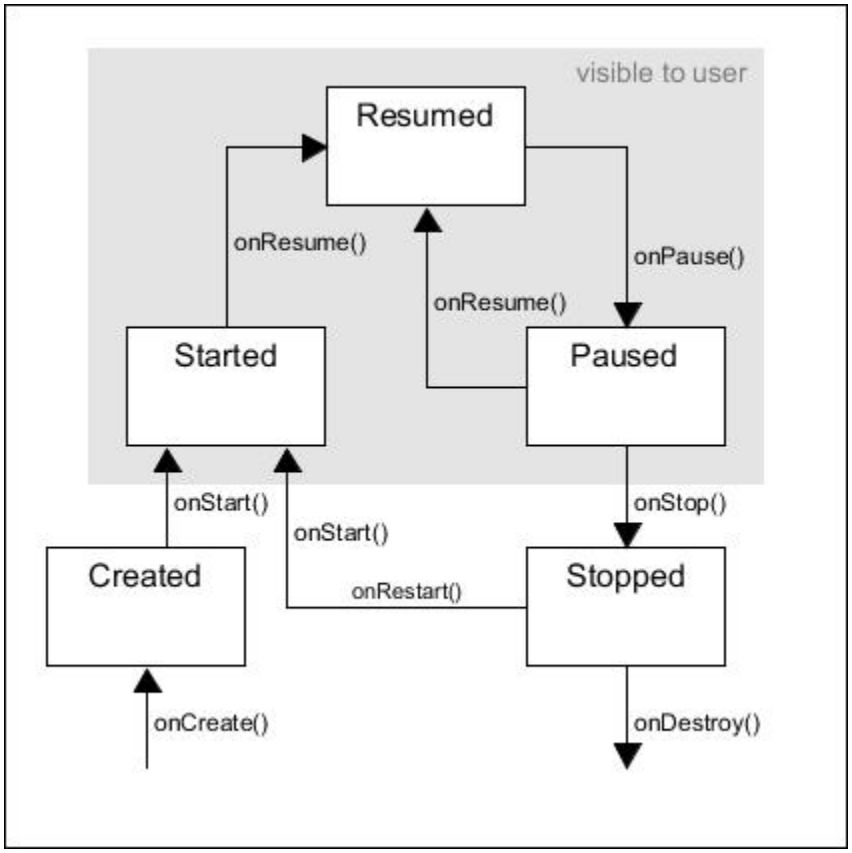

2:35
News Composer
SANDWICH NEWS

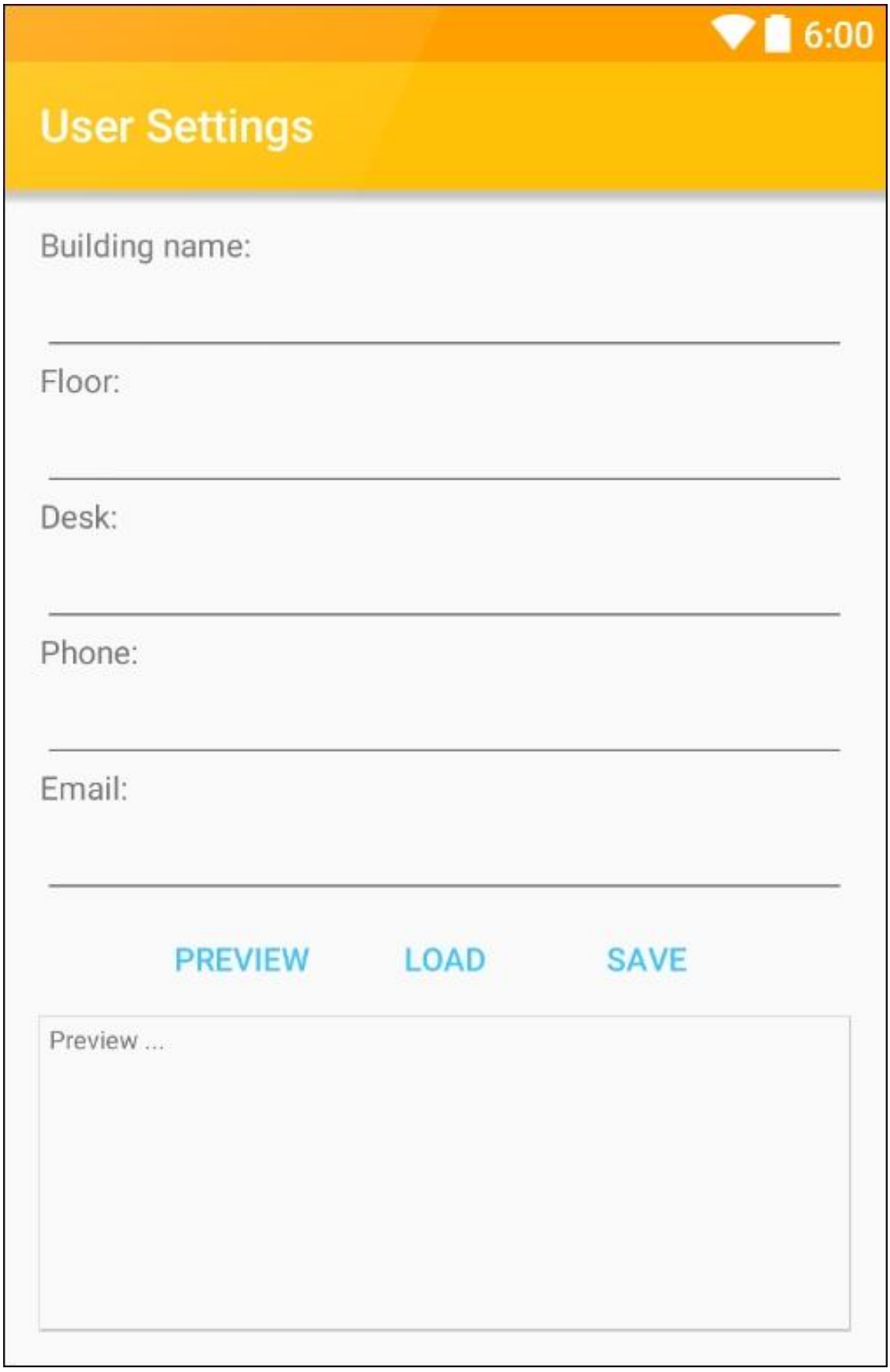

MainActivity
+MainActivity()
#onCreate(Bundle): void
Builder
~header: Component
~headline: Component
~context: Context
~Builder(Context)
-initLeaves(): void
~headerGroup(): Component
~sandwichArray(): Component
~sandwichLayout(): Component
~story(): Component
+setText(int): TextLeaf
+setImage(int): ImageLeaf
«interface»
Component
+add(Component): void
+setContent(int): void
+inflate(ViewGroup): void
ImageLeaf
-imageView: ImageView
+ImageLeaf(ImageView, int)
+add(Component): void
+setContent(int): void
+inflate(ViewGroup): void
+dp(int): int
TextLeaf
+textView: TextView
+TextLeaf(TextView, int)
+add(Component): void
+setContent(int): void
+inflate(ViewGroup): void
-dp(int): int
CompositeImage
~components: List
-imageView: ImageView
+CompositeImage(ImageView, int)
+add(Component): void
+setContent(int): void
+inflate(ViewGroup): void
CompositeText
~components: List
-textView: TextView
+CompositeText(TextView, int)
+add(Component): void
+setContent(int): void
+inflate(ViewGroup): void
CompositeShell
~components: java.util.List
+CompositeShell()
+add(Component): void
+setContent(int): void
+inflate(ViewGroup): void
CompositeLayer
~components: List
-linearLayout: LinearLayout
+CompositeLayer(LinearLayout, int)
+add(Component): void
+setContent(int): void
+inflate(ViewGroup): void

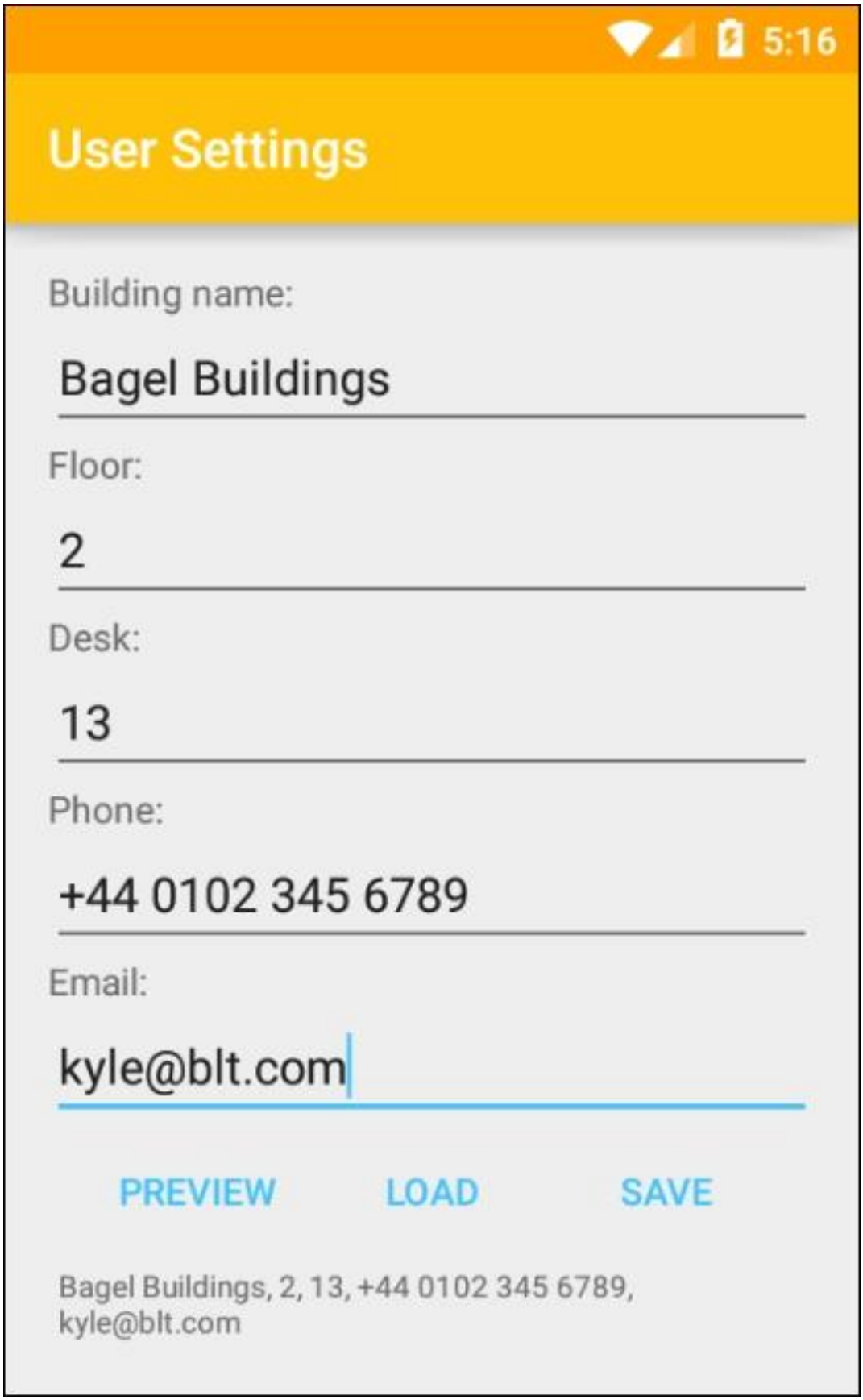

11:33
Static text files
A sandwich is a food item consisting of one or more types of food, such as vegetables, sliced cheese or meat, placed on or between slices of bread, or more generally any dish wherein two or more pieces of bread serve as a container or wrapper for some other food. The sandwich was originally a portable food item or finger food which began to be popular in the Western World. Today sandwiches in various versions are found worldwide. Sandwiches are a popular type of lunch food, taken to work, school, or picnics to be eaten as part of a packed lunch. The bread can be used plain, or it can be coated with one or more condiments such as mayonnaise or mustard to enhance the flavours and texture. As well as being homemade, sandwiches are also widely sold in restaurants and cafes, and are sometimes served hot as well as cold. There are both savoury sandwiches, such as deli meat sandwiches, and sweet sandwiches, such as a peanut butter and jelly sandwich.
11:32
Archivos de texto estático
El sándwich (del inglés sandwich, cuya grafía entró a formar parte de la real academia de la lengua española en 1927), también conocido en español como emparedado, es una comida, a modo de tentempié, aperitivo o la comida que se suele hacer a diario típica de la gastronomía inglesa. Suele consistir en dos rebanadas de pan de molde inglés, o de cualquier tipo de pan, entre las cuales se coloca otros alimentos tales como carne, queso, verduras u otros, a veces con condimentos, salsas u otros acompañantes. Cabe destacar que en el Reino Unido, el clásico sándwich de pan de molde inglés se está siendo sustituido por otros tipos de pan, así, al sándwich de pan francés, lo denominan French bread sandwich o French baguette sandwich'


Component Tree
Device Screen
RelativeLayout
text_view (TextView) - @string/prompt
edit_text (EditText)
LinearLayout (horizontal)
action_load (Button) - @string/action_load
action_save (Button) - @string/action_save

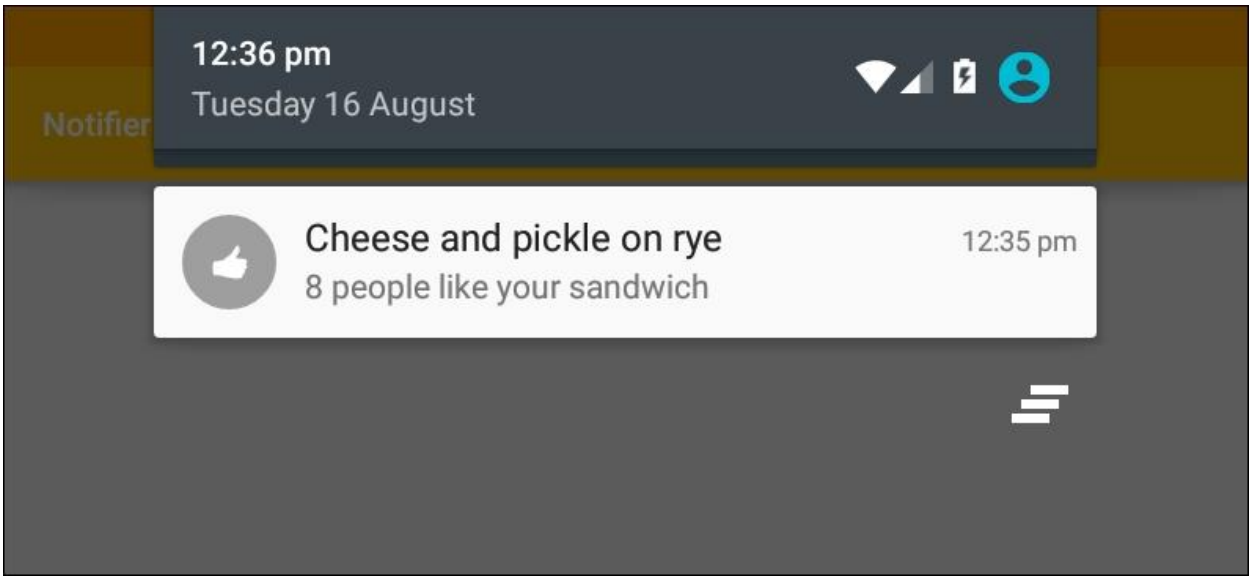

4:47
Dynamic Files
Enter your comment...
During the Middle Ages in Europe, thick slabs of coarse and usually stale bread, called "trenchers", were used as plates. After a meal, the food soaked trencher was fed to a dog or to beggars at the tables of the wealthy and eaten by diners in more modest circumstances.
LOAD
SAVE

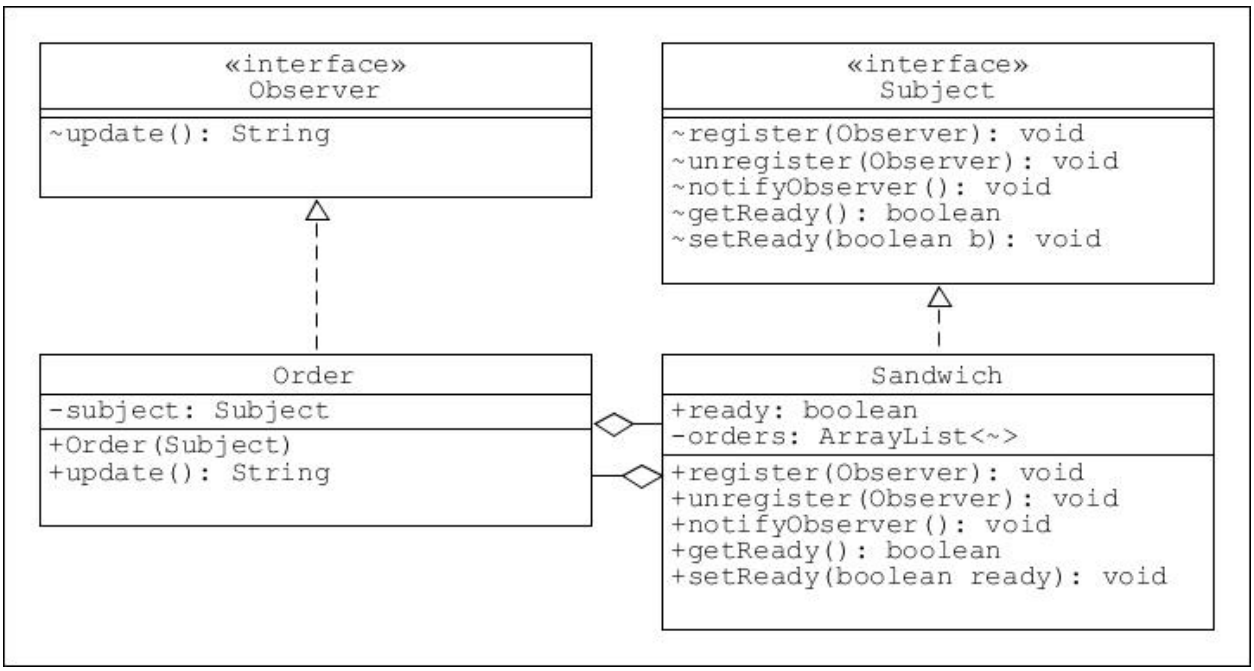

visible to user
Resumed
onResume()
onPause()
onResume()
Started
Paused
onStart()
onStart()
onStop()
Created
onRestart()
Stopped
onCreate()
onDestroy()

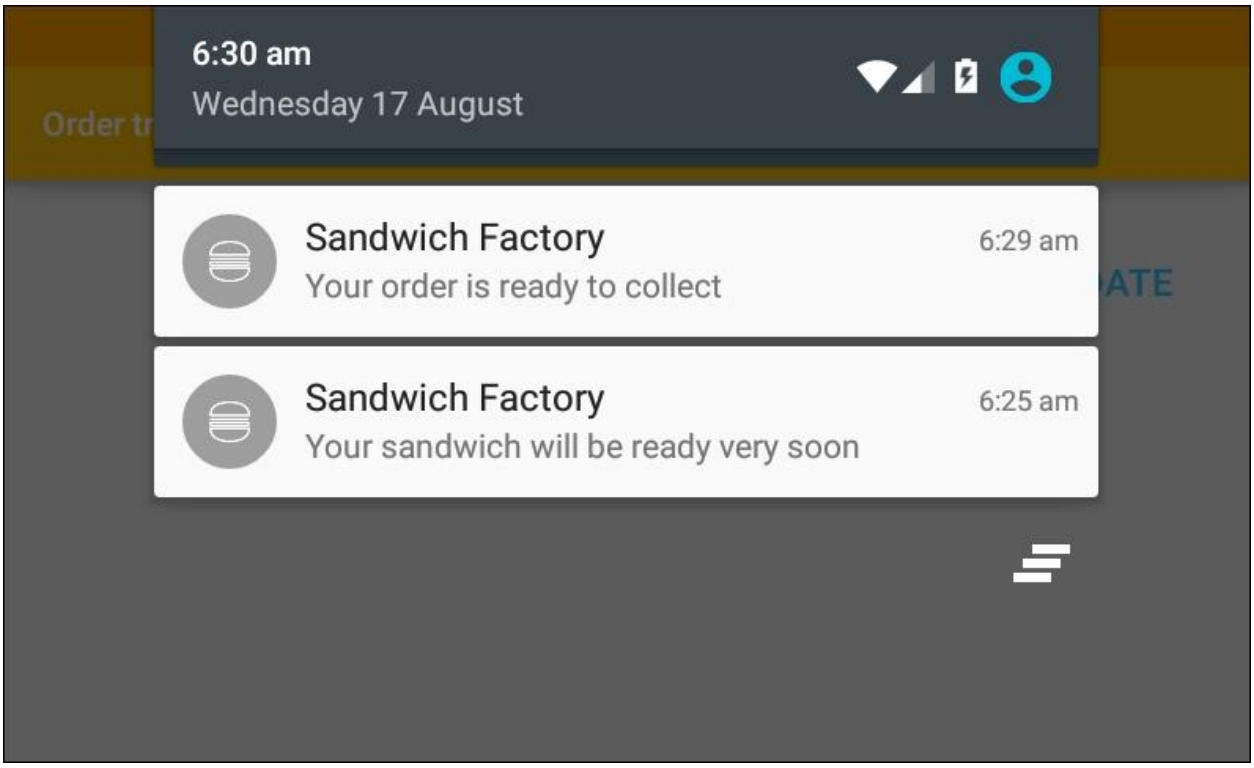

6:00
User Settings
Building name:
Floor:
Desk:
Phone:
Email:
PREVIEW
LOAD
SAVE
Preview ...

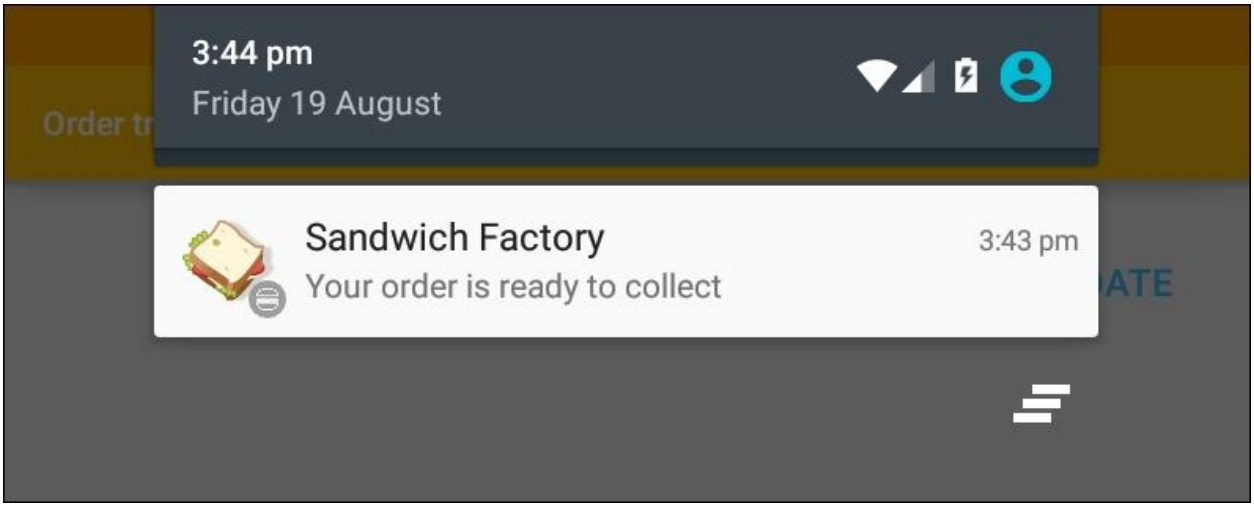

5:16
User Settings
Building name:
Bagel Buildings
Floor:
2
Desk:
13
Phone:
+44 0102 345 6789
Email:
kyle@blt.com
PREVIEW
LOAD
SAVE
Bagel Buildings, 2, 13, +44 0102 345 6789,
kyle@blt.com

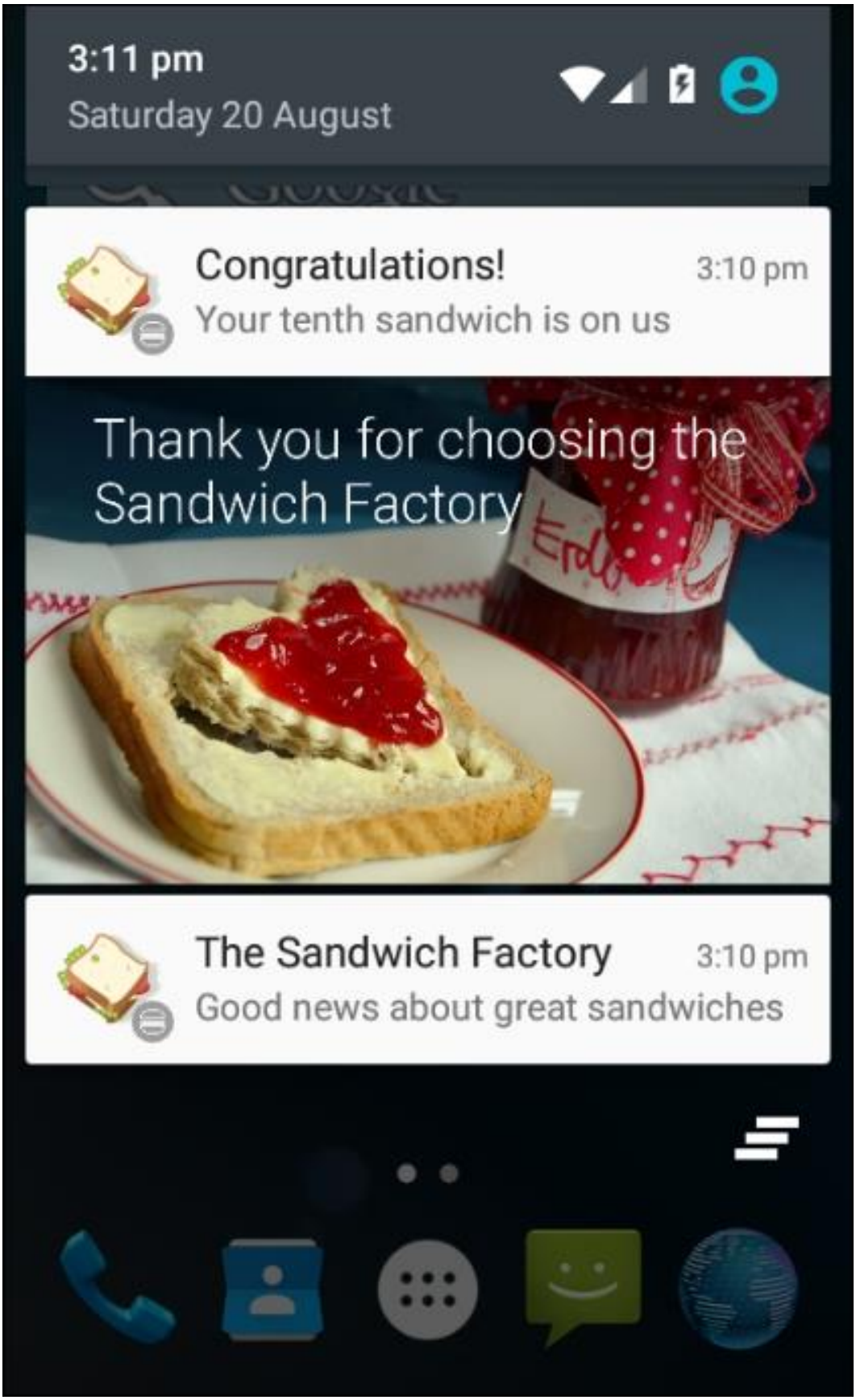
# Chapter 9: Observing Patterns

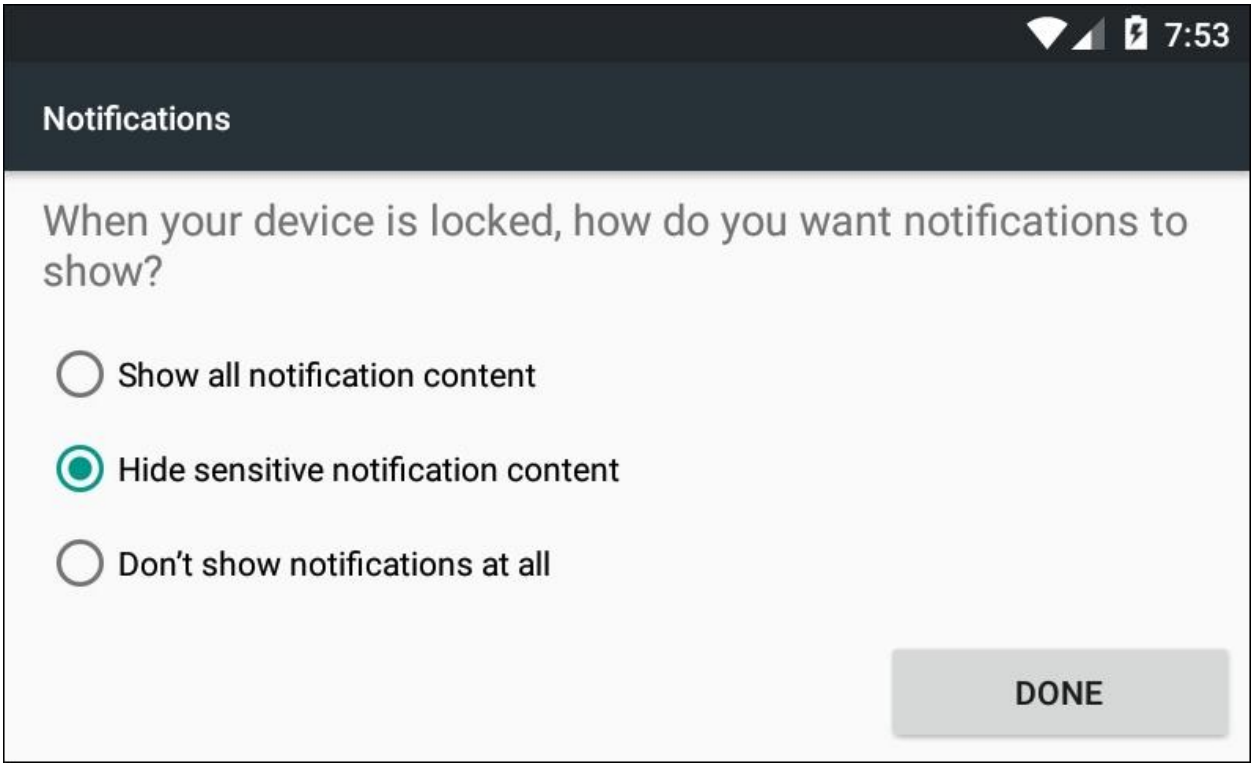


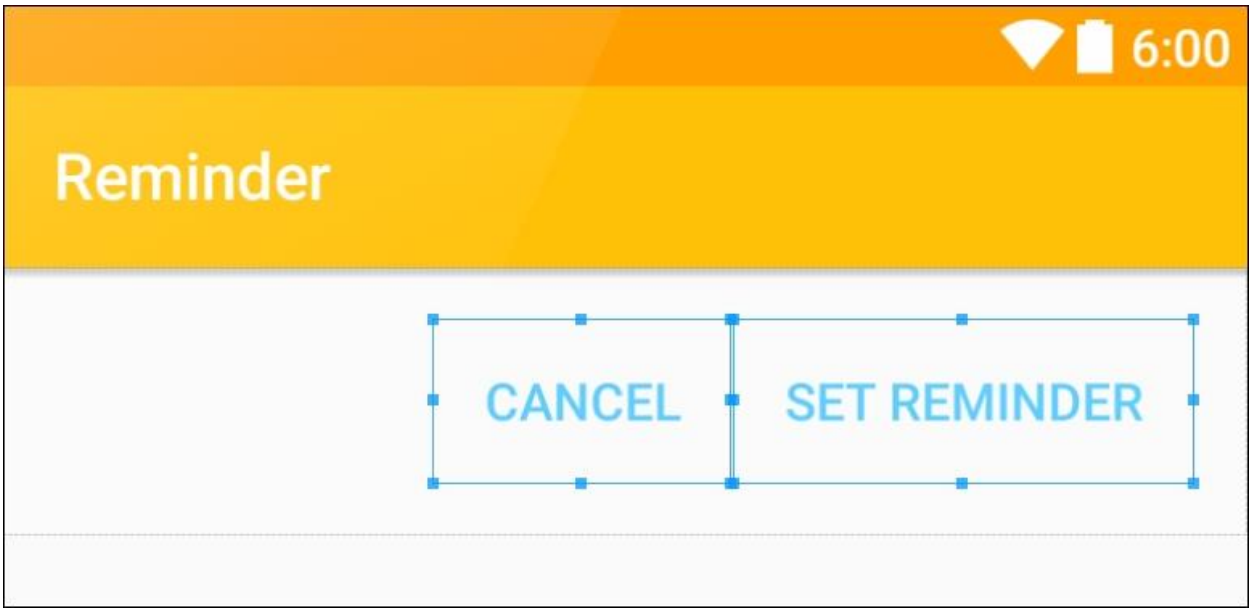




6:30 am
Wednesday 17 August
Sandwich Factory
6:29 am
Your order is ready to collect
Sandwich Factory
6:25 am
Your sandwich will be ready very soon

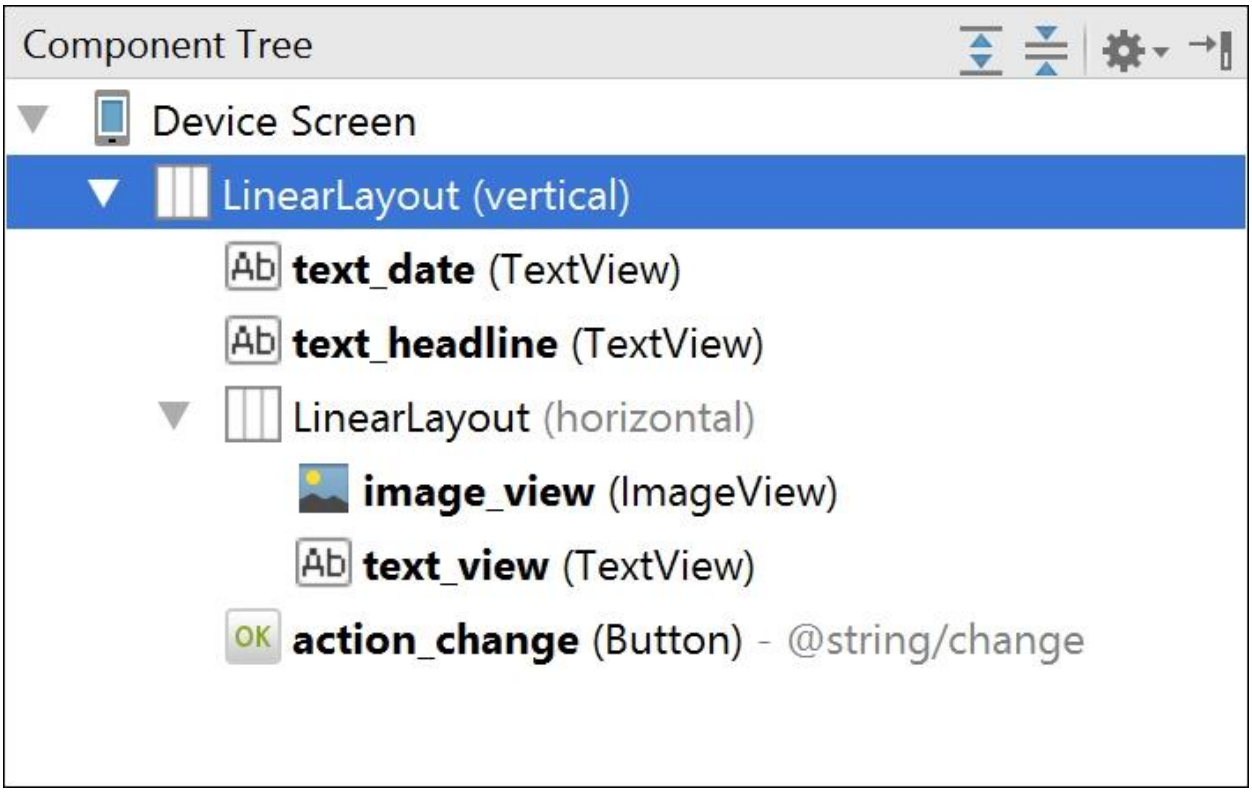

3:44 pm
Friday 19 August
Sandwich Factory
3:43 pm
Your order is ready to collect


3:11 pm
Saturday 20 August
Congratulations!
3:10 pm
Your tenth sandwich is on us
Thank you for choosing the Sandwich Factory
The Sandwich Factory
3:10 pm
Good news about great sandwiches


7:53
Notifications
When your device is locked, how do you want notifications to show?
Show all notification content
Hide sensitive notification content
Don’t show notifications at all
DONE


6:00
Reminder
CANCEL
SET REMINDER

## Chapter 10: Behavioral Patterns



10:34

NEWS TODAY

September 2

## World's Tastiest Sandwich

The verb to sandwich has the meaning to position anything between two other things of a different character, or to place different elements alternately.

CHANGE

10:34

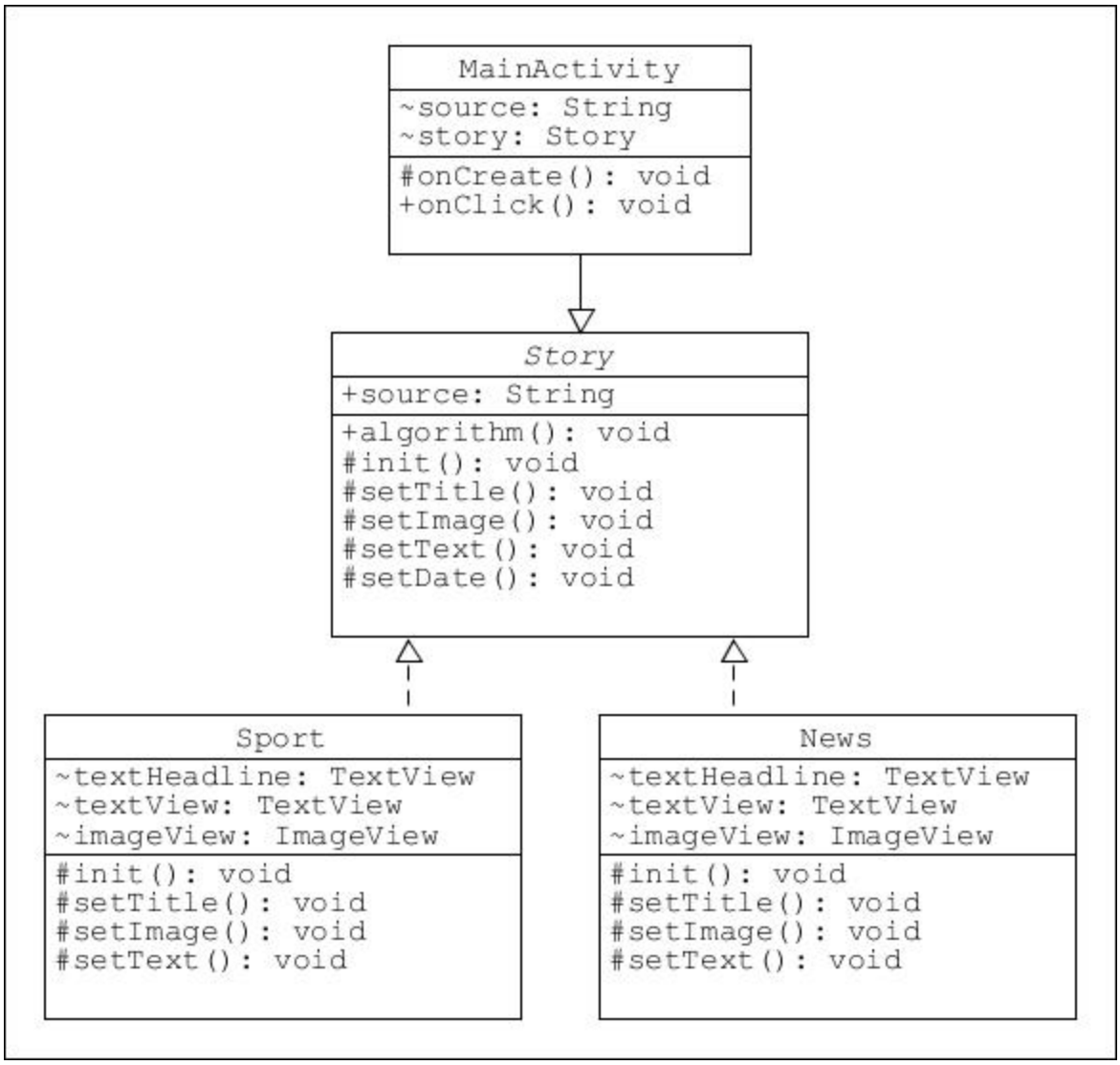
SPORT TODAY

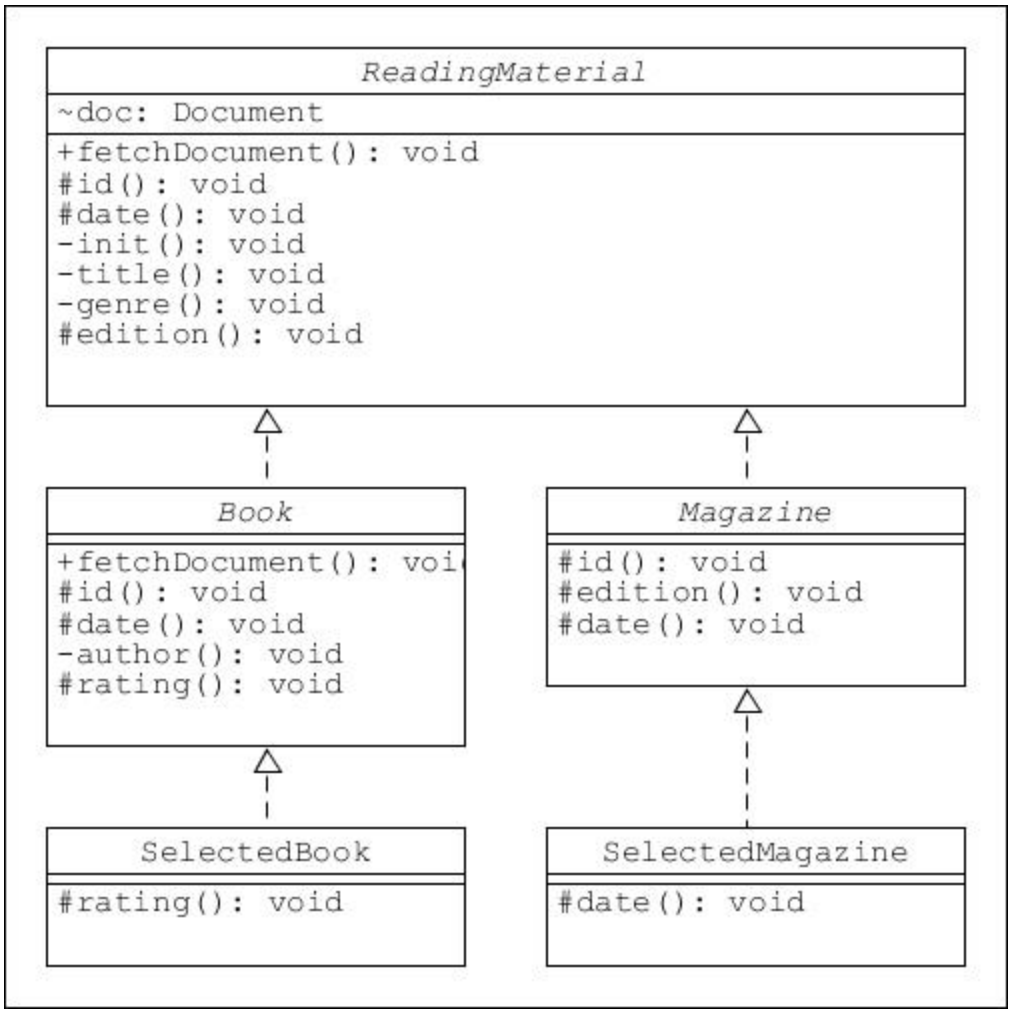
September 2

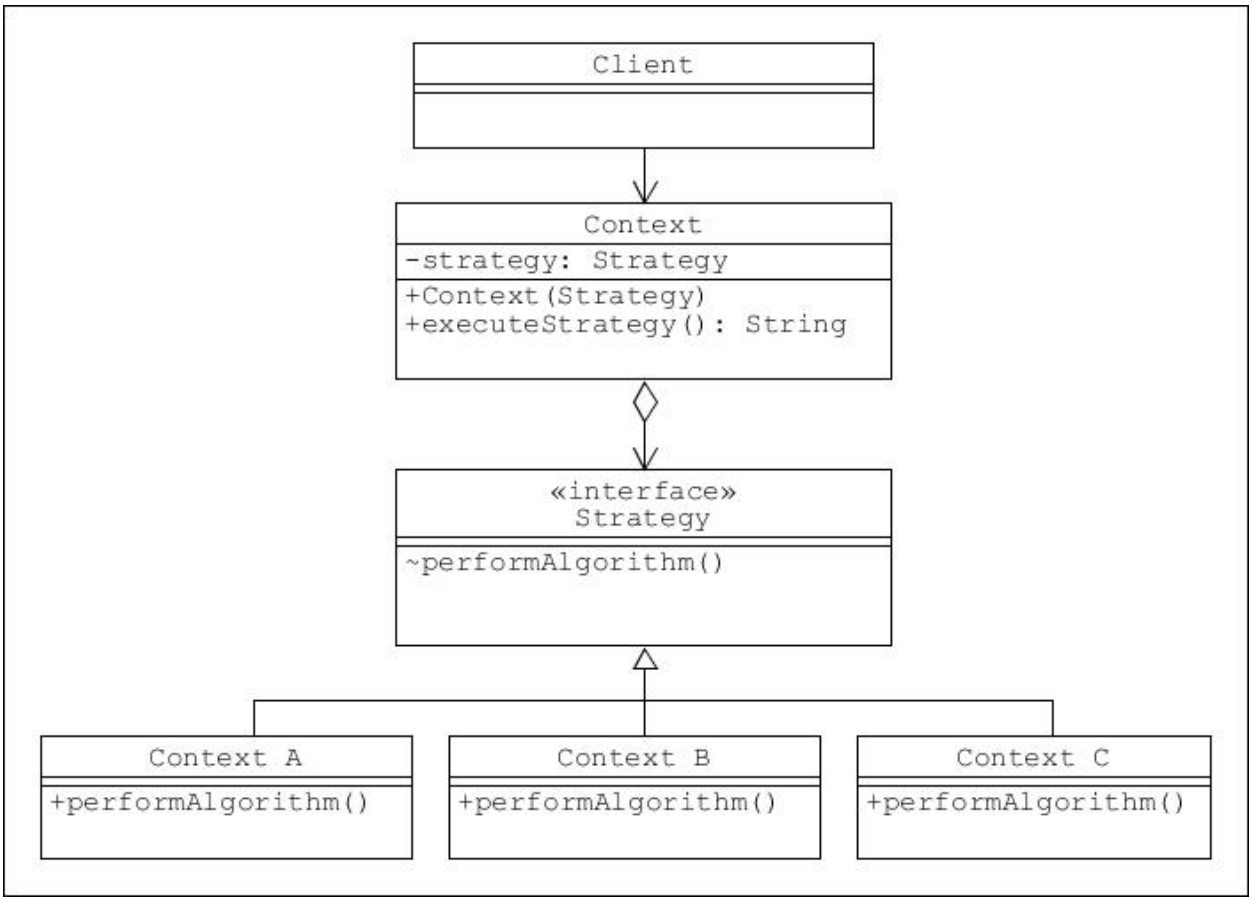
## Sports News Today

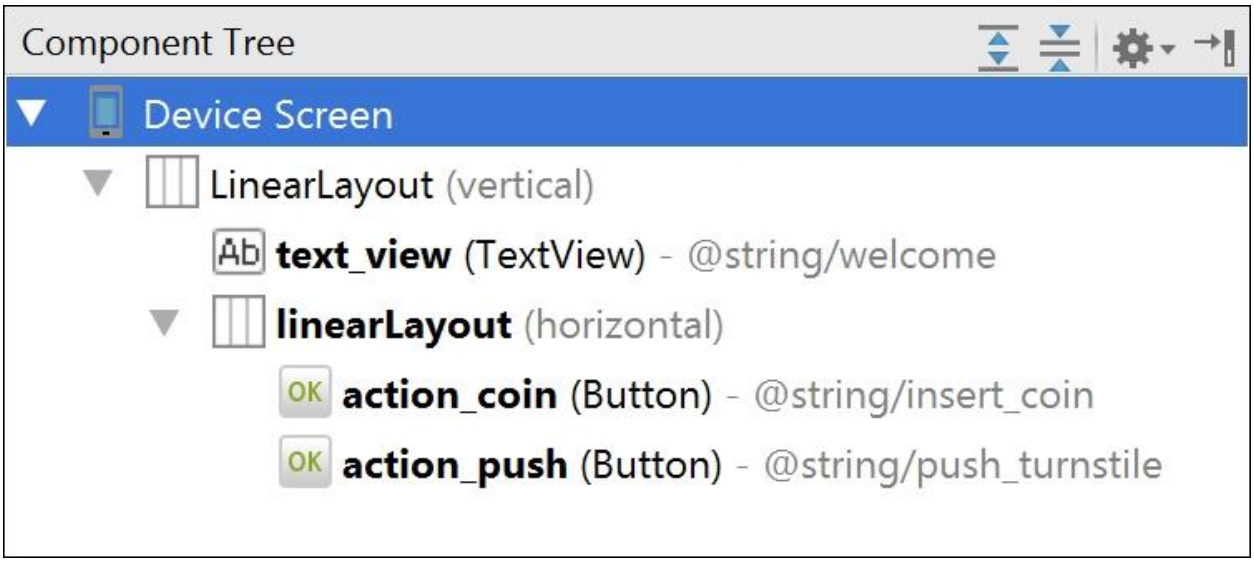
Sports are all forms of usually competitive physical activity or games which, through casual or organised participation, aim to use, maintain or

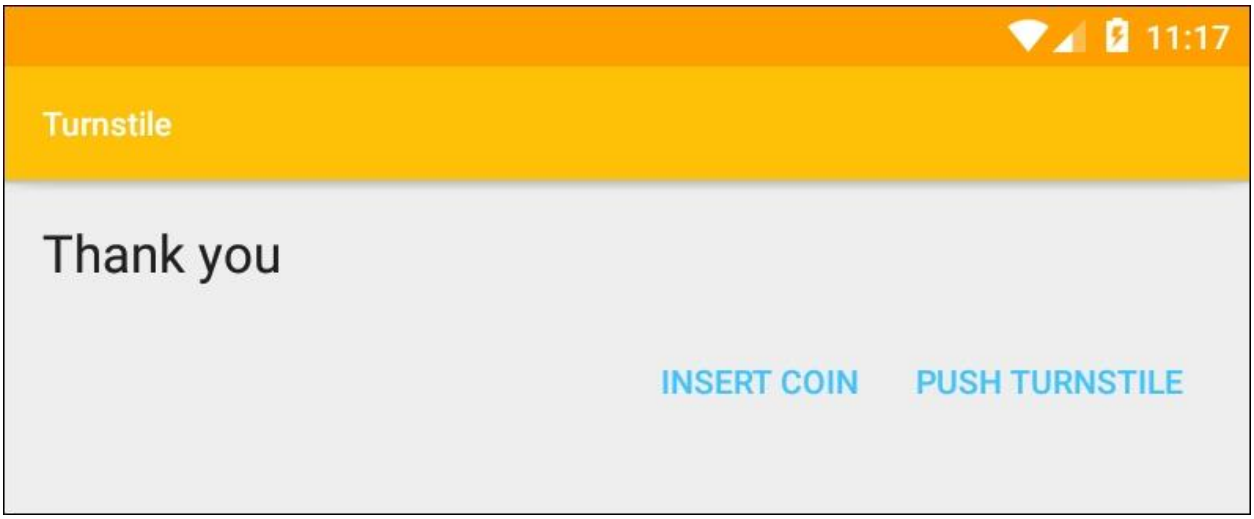
CHANGE

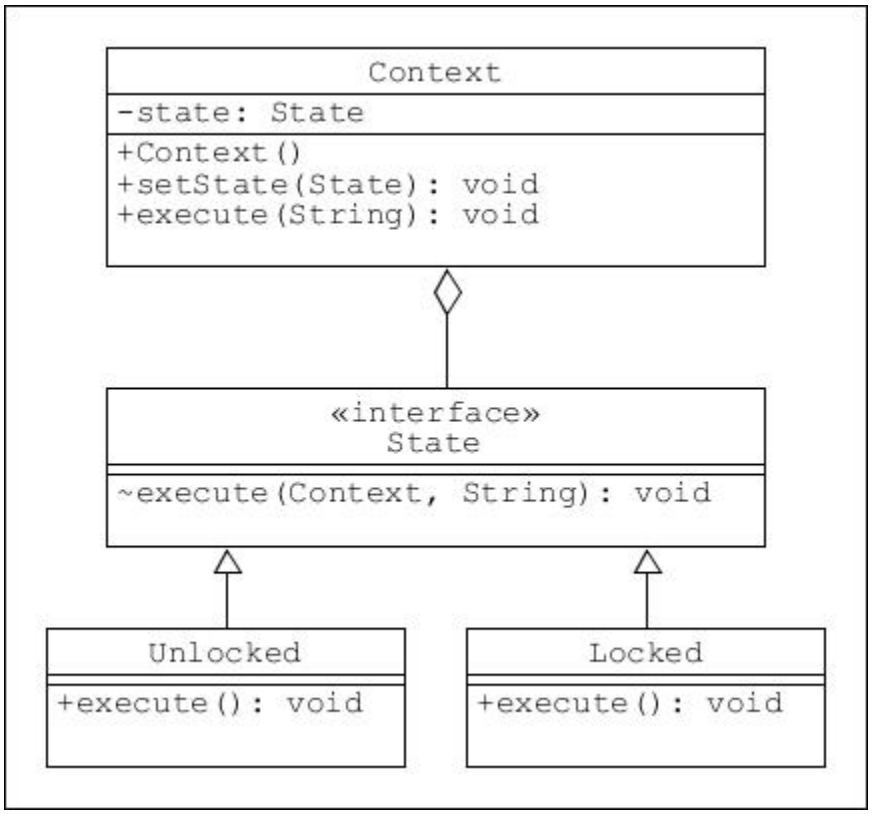

MainActivity
~source: String
~story: Story
#onCreate(): void
+onClick(): void
Story
+source: String
+algorithm(): void
#init(): void
#setTitle(): void
#setImage(): void
#setText(): void
#setDate(): void
Sport
~textHeadline: TextView
~textView: TextView
~imageView: ImageView
#init(): void
#setTitle(): void
#setImage(): void
#setText(): void
News
~textHeadline: TextView
~textView: TextView
~imageView: ImageView
#init(): void
#setTitle(): void
#setImage(): void
#setText(): void


ReadingMaterial
~doc: Document
+fetchDocument(): void
#id(): void
#date(): void
-init(): void
-title(): void
-genre(): void
#edition(): void
Book
+fetchDocument(): voi
#id(): void
#date(): void
-author(): void
#rating(): void
Magazine
#id(): void
#edition(): void
#date(): void
SelectedBook
#rating(): void
SelectedMagazine
#date(): void

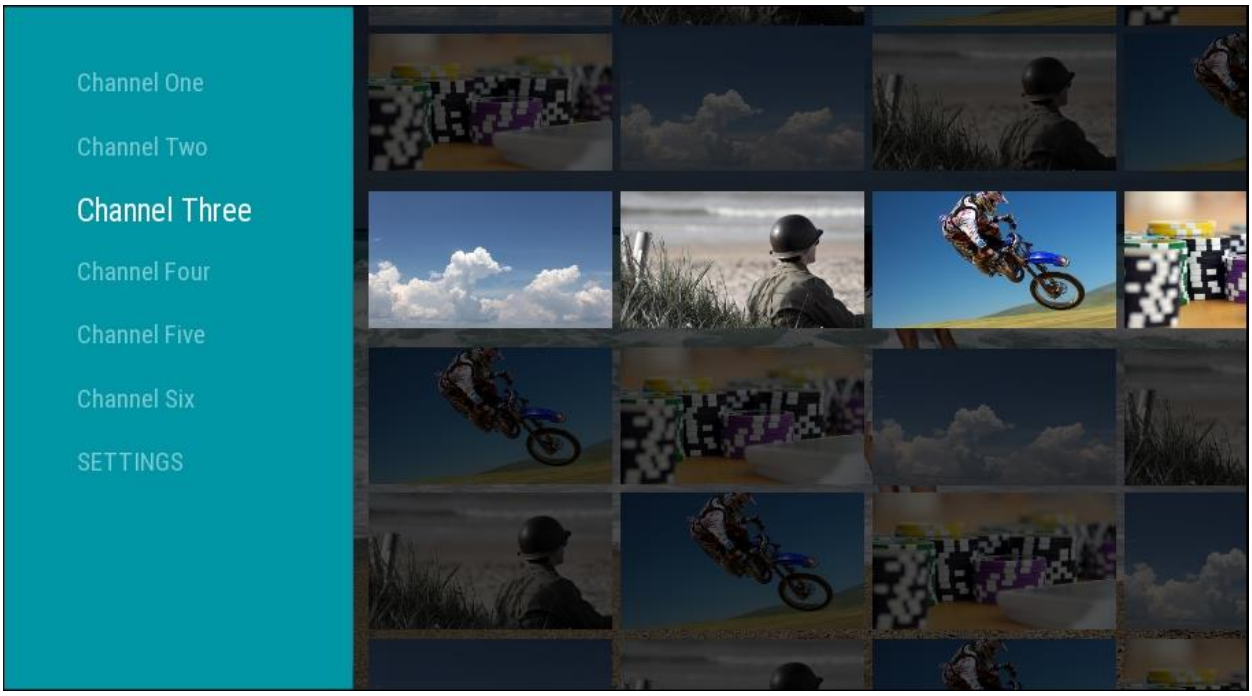

Client
Context
-strategy: Strategy
+Context(Strategy)
+executeStrategy(): String
«interface»
Strategy
~performAlgorithm()
Context A
+performAlgorithm()
Context B
+performAlgorithm()
Context C
+performAlgorithm()

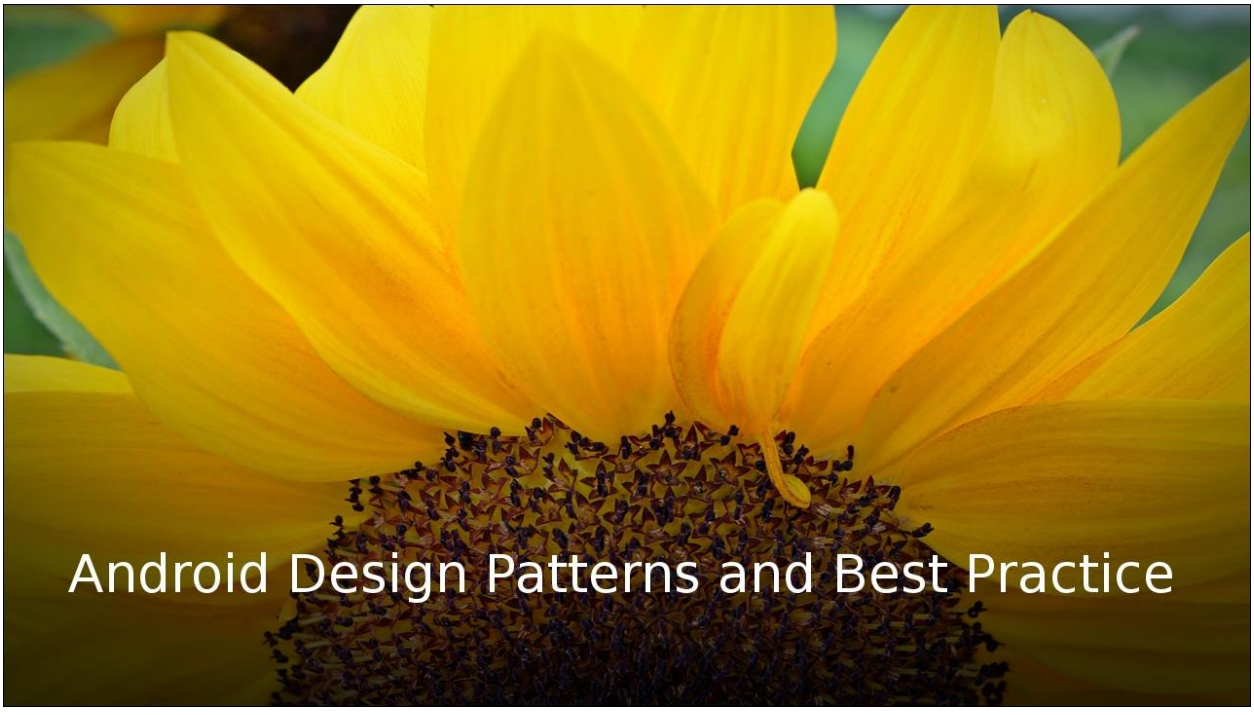

Component Tree
Device Screen
LinearLayout (vertical)
text_view (TextView) - @string/welcome
linearLayout (horizontal)
action_coin (Button) - @string/insert_coin
action_push (Button) - @string/push_turnstile

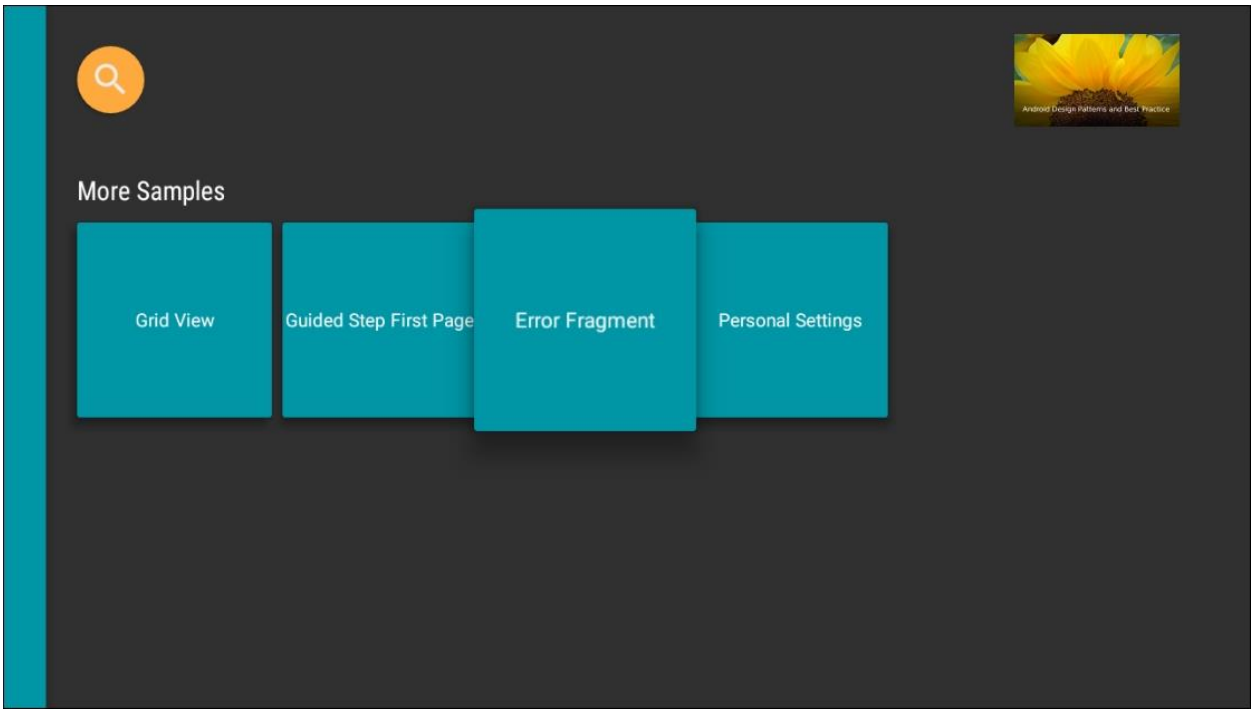

11:17
Turnstile
Thank you
INSERT COIN
PUSH TURNSTILE

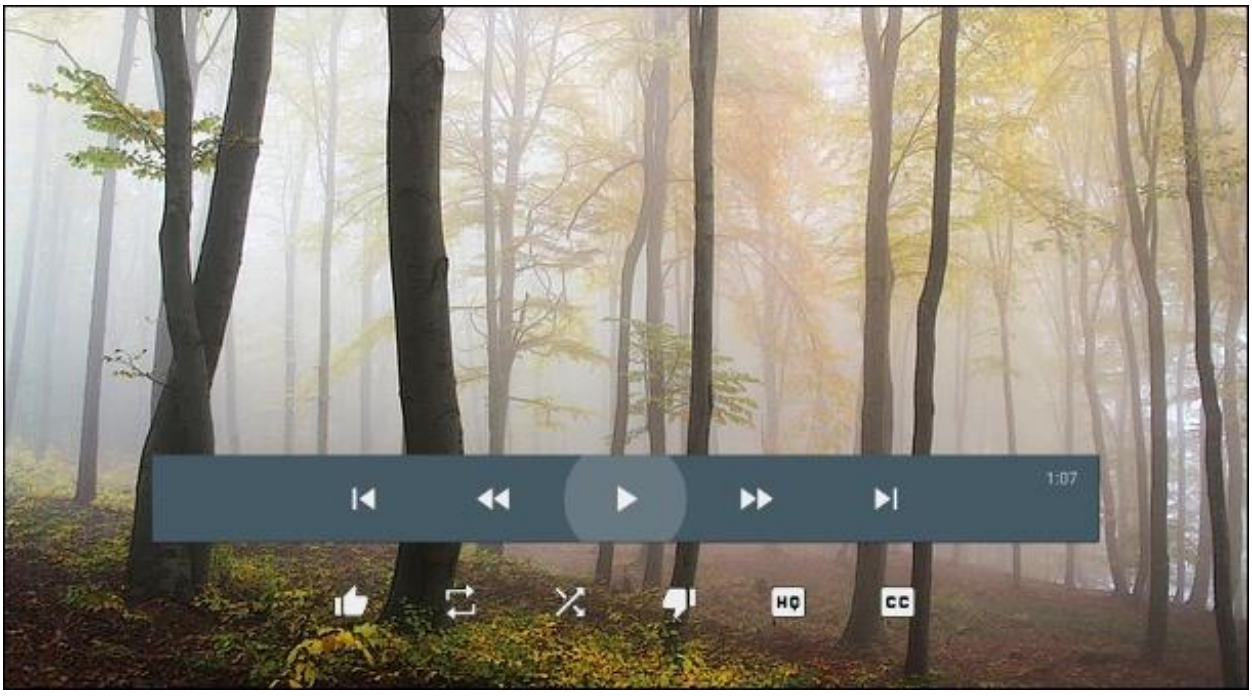

Context
-state: State
+Context()
+setState(State): void
+execute(String): void
«interface»
State
~execute(Context, String): void
Unlocked
+execute(): void
Locked
+execute(): void

# Chapter 11: Wearable Patterns


Channel One
Channel Two
Channel Three
Channel Four
Channel Five
Channel Six
SETTINGS


Android Design Patterns and Best Practice

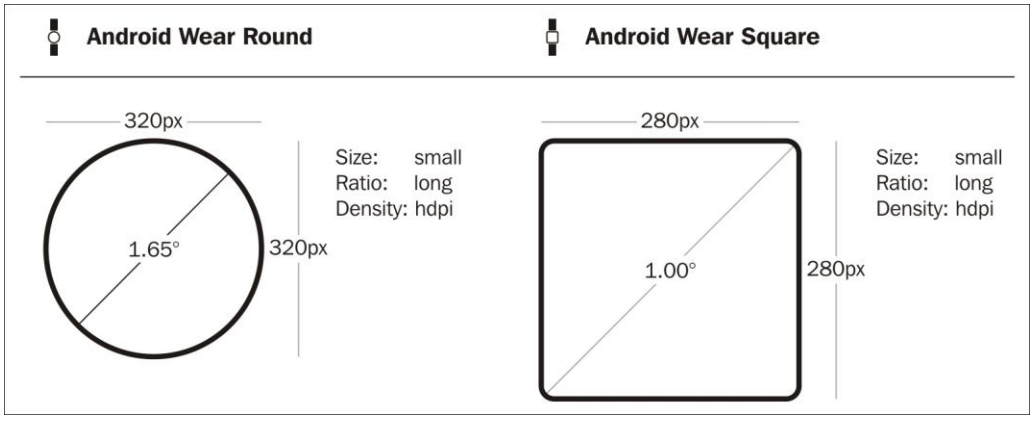

Android Design Patterns and Best Practice
More Samples
Grid View
Guided Step First Page
Error Fragment
Personal Settings

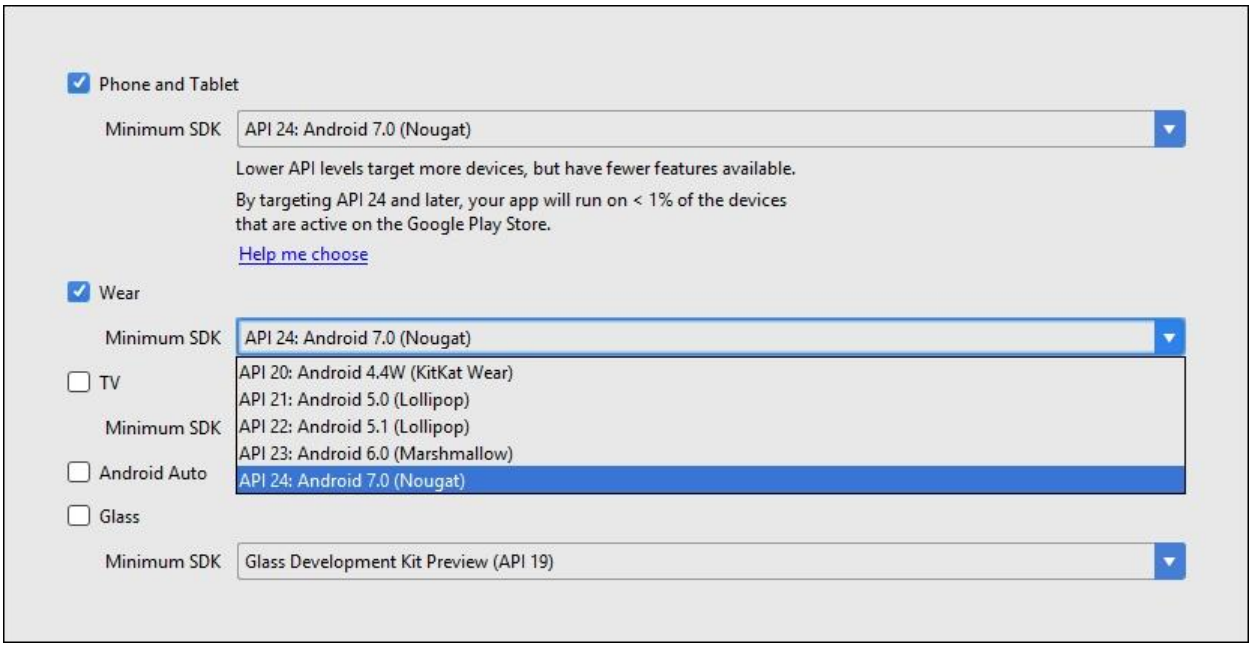

1:07
HQ
CC

# The Movie II

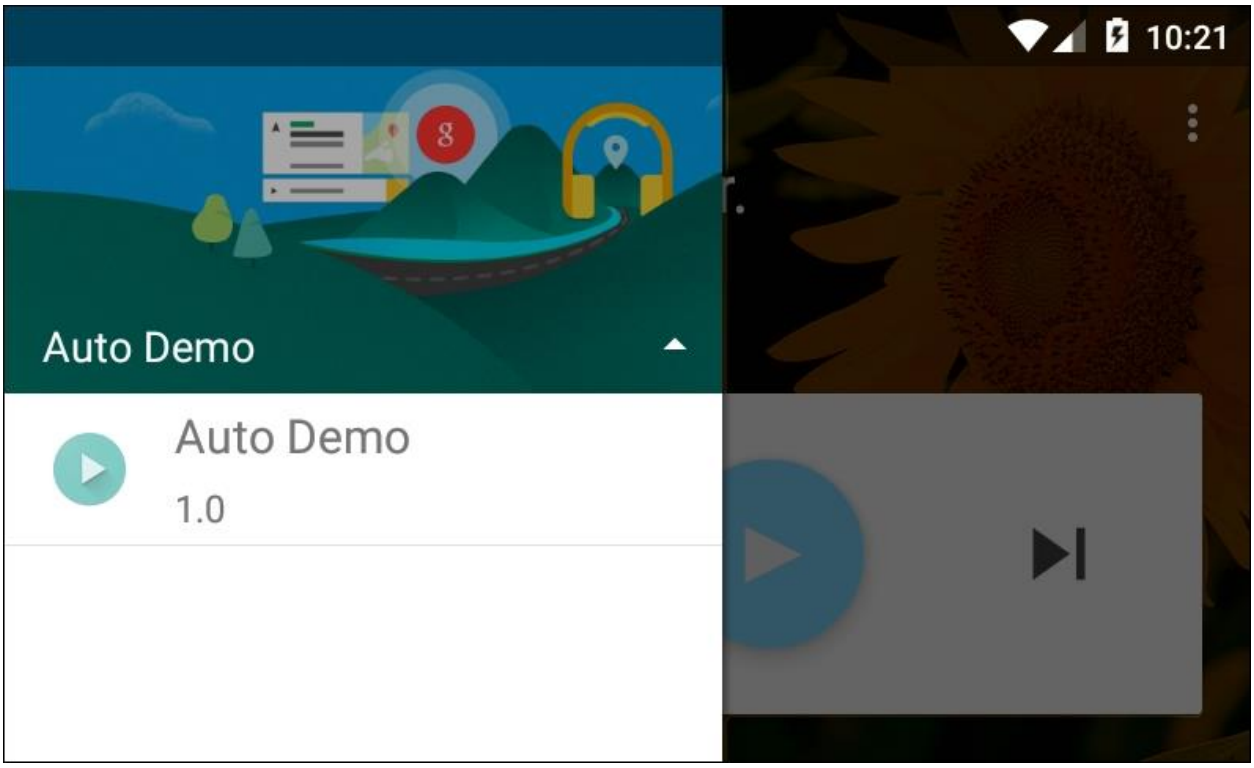
The sequel to The Movie

The Movie II
The sequel to The Movie

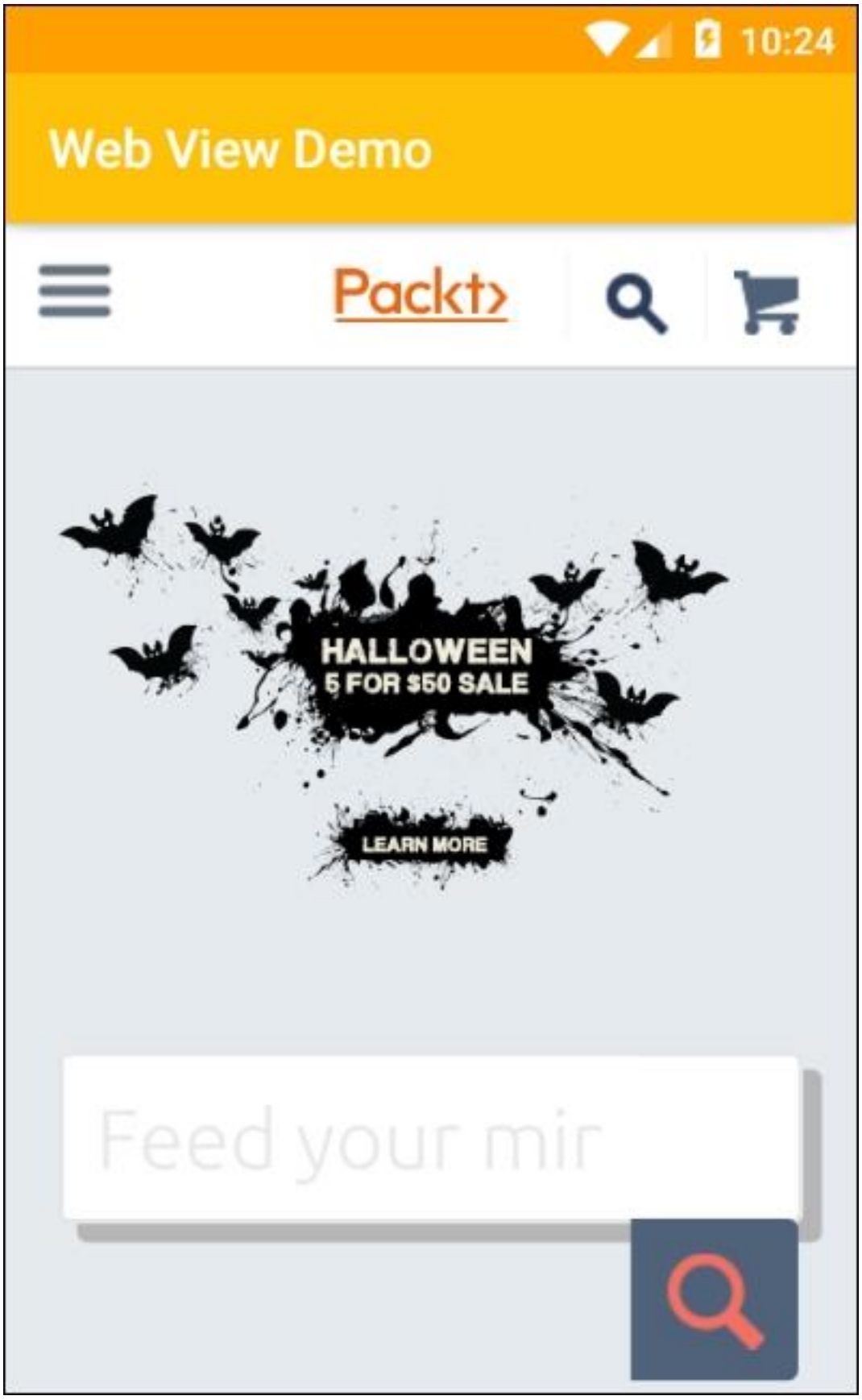

Android Wear Round
320px
1.65°
320px
Size: small
Ratio: long
Density: hdpi
Android Wear Square
280px
1.00°
280px
Size: small
Ratio: long
Density: hdpi

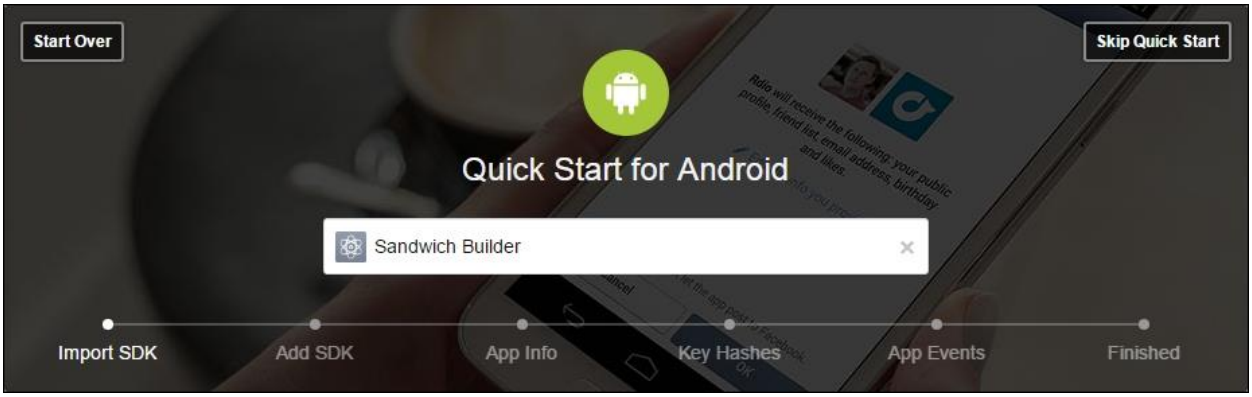

Phone and Tablet
Minimum SDK
API 24: Android 7.0 (Nougat)
Lower API levels target more devices, but have fewer features available.
By targeting API 24 and later, your app will run on < 1% of the devices that are active on the Google Play Store.
Help me choose
Wear
Minimum SDK
API 24: Android 7.0 (Nougat)
TV
Minimum SDK
Android Auto
Glass
Minimum SDK
Glass Development Kit Preview (API 19)
API 20: Android 4.4W (KitKat Wear)
API 21: Android 5.0 (Lollipop)
API 22: Android 5.1 (Lollipop)
API 23: Android 6.0 (Marshmallow)
API 24: Android 7.0 (Nougat)

74.0 bpm


10:21
Auto Demo
Auto Demo
1.0

# Chapter 12: Social Patterns


10:24
Web View Demo
Packt›
HALLOWEEN
5 FOR $50 SALE
LEARN MORE
Feed your mir




## Create a New App ID

Get started integrating Facebook into your app or website

Display Name

Sandwich Builder

Contact Email

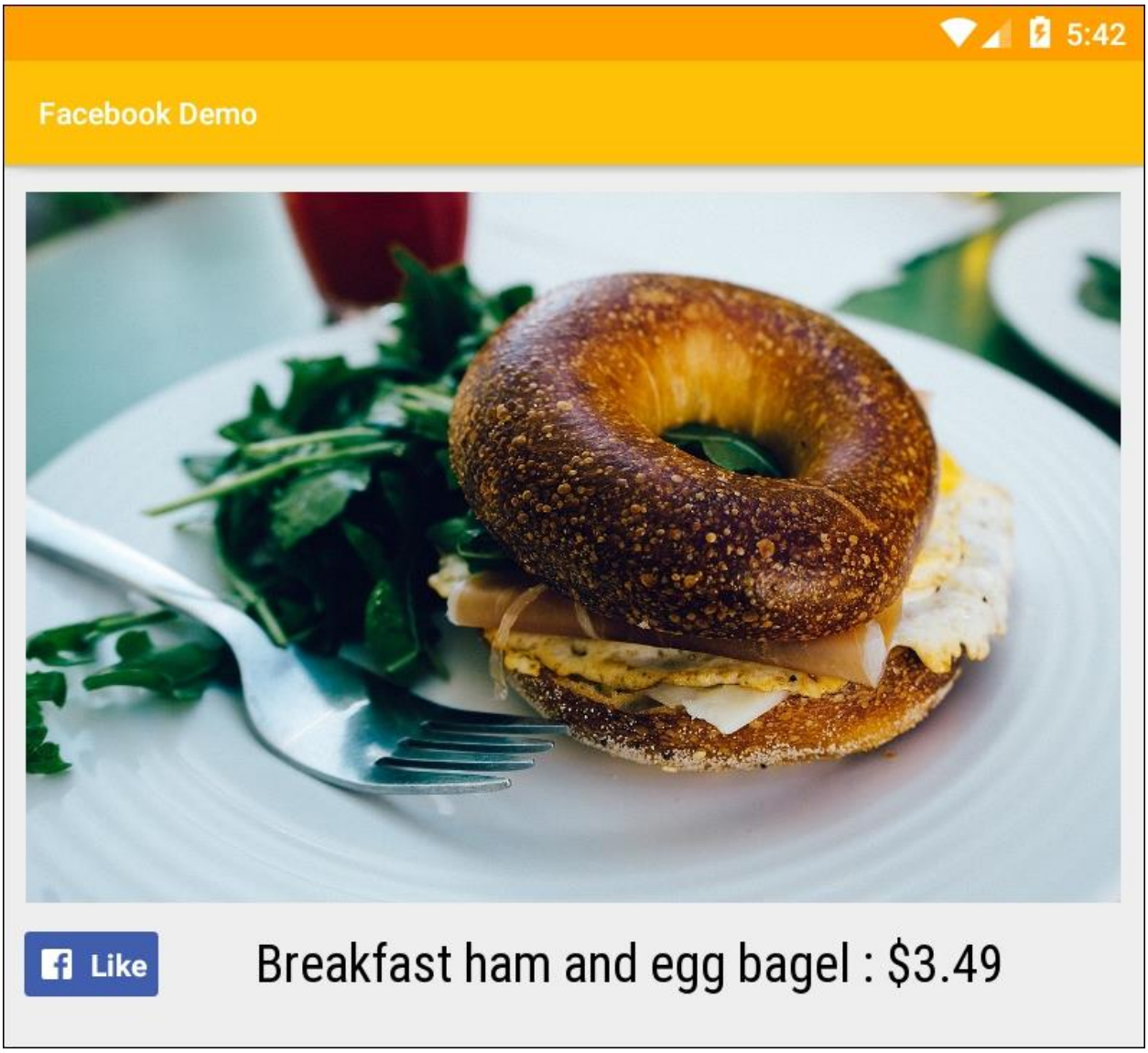
Used for important communication about your app

Category

Food & Drink

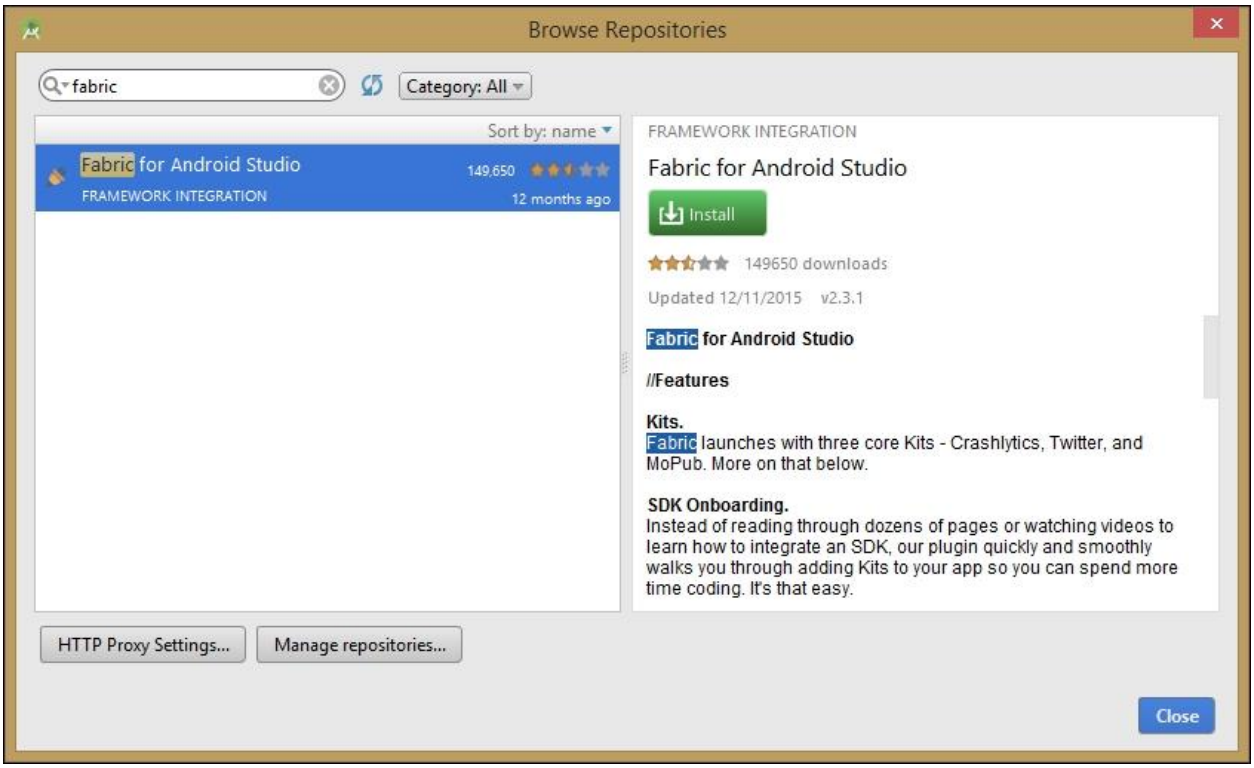
By proceeding, you agree to the Facebook Platform Policies

Cancel | Create App ID


5:42
Facebook Demo
Like
Breakfast ham and egg bagel : $3.49

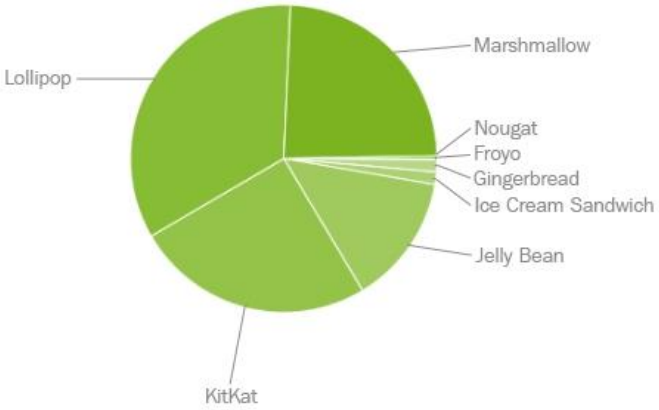
**When and how we last scraped the URL**

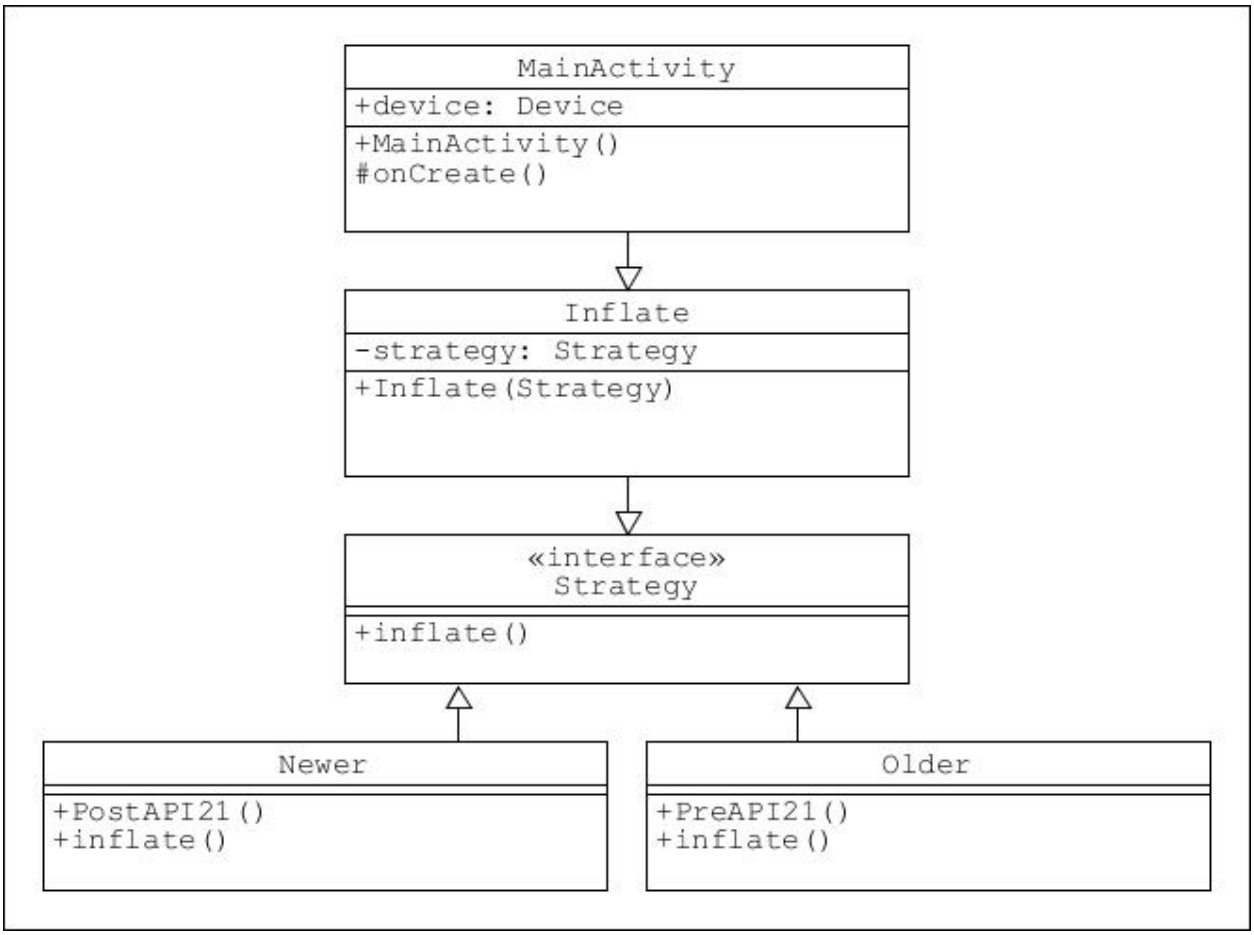
| | |
|---|---|
| Time Scraped | 8 November at 22:17 Scrape Again |
| Response Code | 200 |
| Fetched URL | https://developers.facebook.com/docs/sharing/android |
| Canonical URL | https://developers.facebook.com/docs/sharing/android<br>312 likes, shares and comments (More Info) |
| Link Preview | <br>Android - Sharing - Documentation - Facebook for Developers<br>Sharing. Guide assumes a developer will use the solution, already started designing their product, or is already implementing it.<br>DEVELOPERS.FACEBOOK.COM |

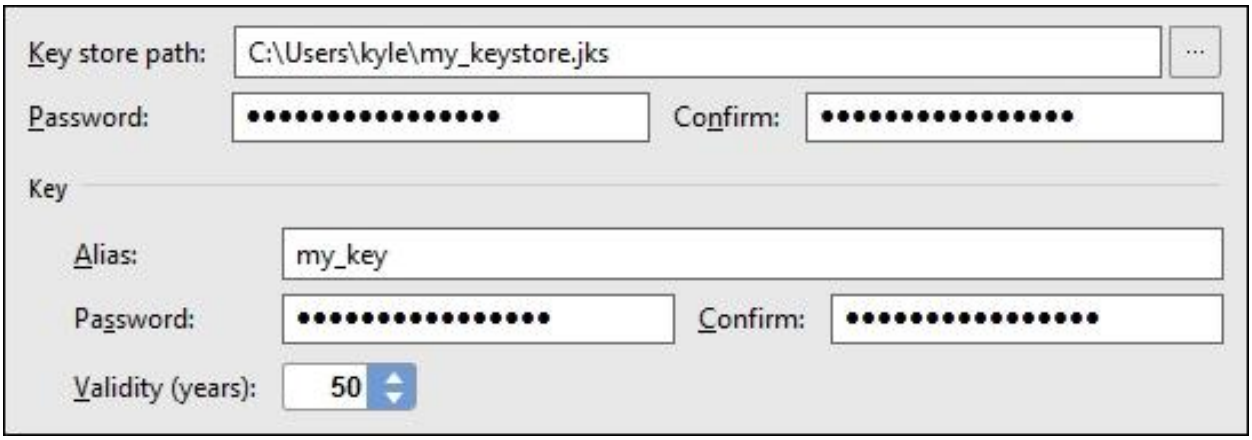


# Chapter 13: Distribution Patterns

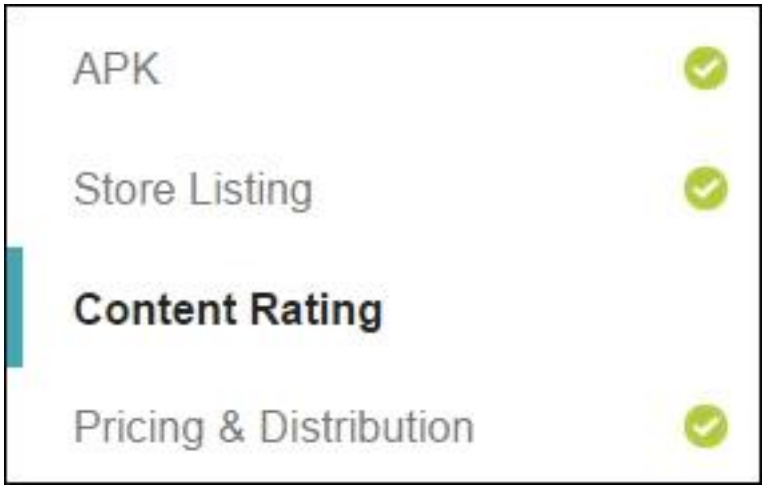
| Version | Codename | API | Distribution |
|---|---|---|---|
| 2.2 | Froyo | 8 | 0.1% |
| 2.3.3 - 2.3.7 | Gingerbread | 10 | 1.3% |
| 4.0.3 - 4.0.4 | Ice Cream Sandwich | 15 | 1.3% |
| 4.1.x | Jelly Bean | 16 | 4.9% |
| 4.2.x | | 17 | 6.8% |
| 4.3 | | 18 | 2.0% |
| 4.4 | KitKat | 19 | 25.2% |
| 5.0 | Lollipop | 21 | 11.3% |
| 5.1 | | 22 | 22.8% |
| 6.0 | Marshmallow | 23 | 24.0% |
| 7.0 | Nougat | 24 | 0.3% |




MainActivity
+device: Device
+MainActivity()
#onCreate()
Inflate
-strategy: Strategy
+Inflate(Strategy)
«interface»
Strategy
+inflate()
Newer
+PostAPI21()
+inflate()
Older
+PreAPI21()
+inflate()


Key store path:
C:\Users\kyle\my_keystore.jks
...
Password:
Confirm:
Key
Alias:
my_key
Password:
Confirm:
Validity (years):
50

Developer Name
Will appear to users under the name of your application
Email Address
Website URL
Phone Number
Include plus sign, country code and area code. For example, +1-650-253-0000.
Email Updates
Contact me occasionally about development and Google Play opportunities.


APK
Store Listing
Content Rating
Pricing & Distribution

ANDROID APP ON
Google play
ANDROID APP ON
Google play
GET IT ON
Google play
GET IT ON
Google play

Brie

ADD

Camembert

ADD

320×50
Test Banner
AdMob by Google