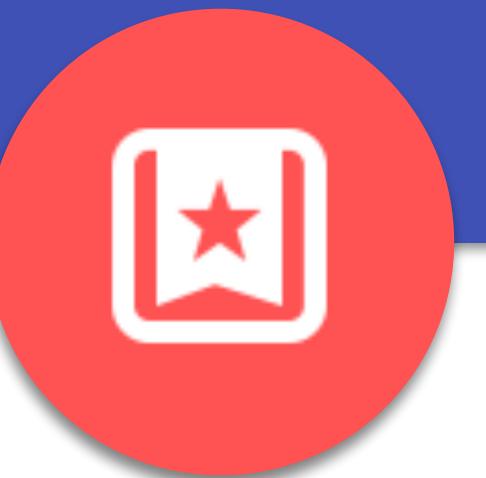


'Materializing' Wunderlist for Android

Material Fest @ Cádiz - 01/02/15

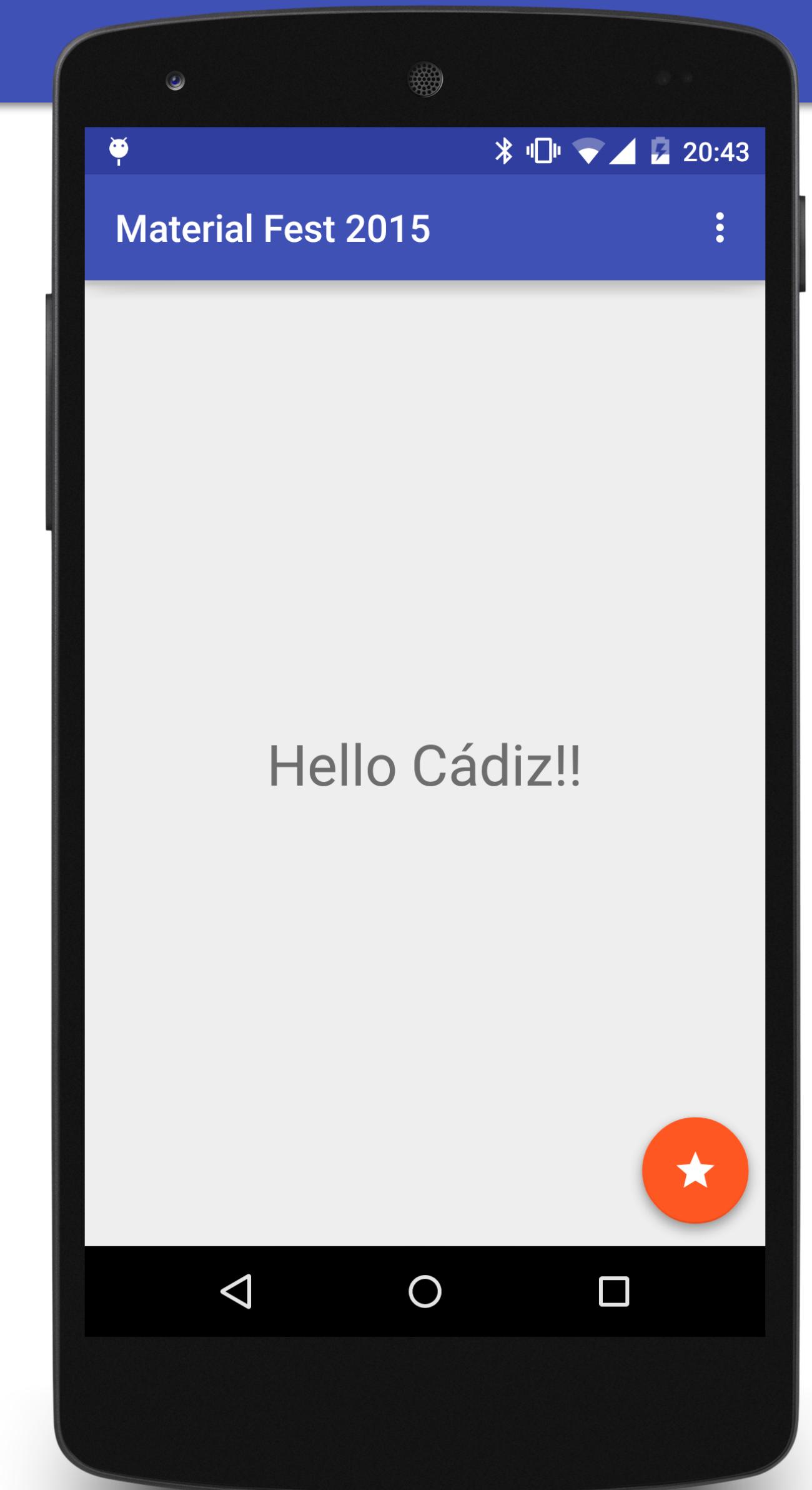


Antonio J. Consuegra
@aconsuegra

6Wunderkinder GmbH

'Materializing' Wunderlist for Android

- Finally, design guidelines from Google.
- Tasty candy for designers.
- Wunderlist has a strong design character.
- Redesign everything? Or keep our own style?
- Let's see :)



Tasks View - Toolbar

- Generalization of ActionBar.
- 48dip vs 56dip.
- Default elevation 4dip.
- More flexible, but more work needed.



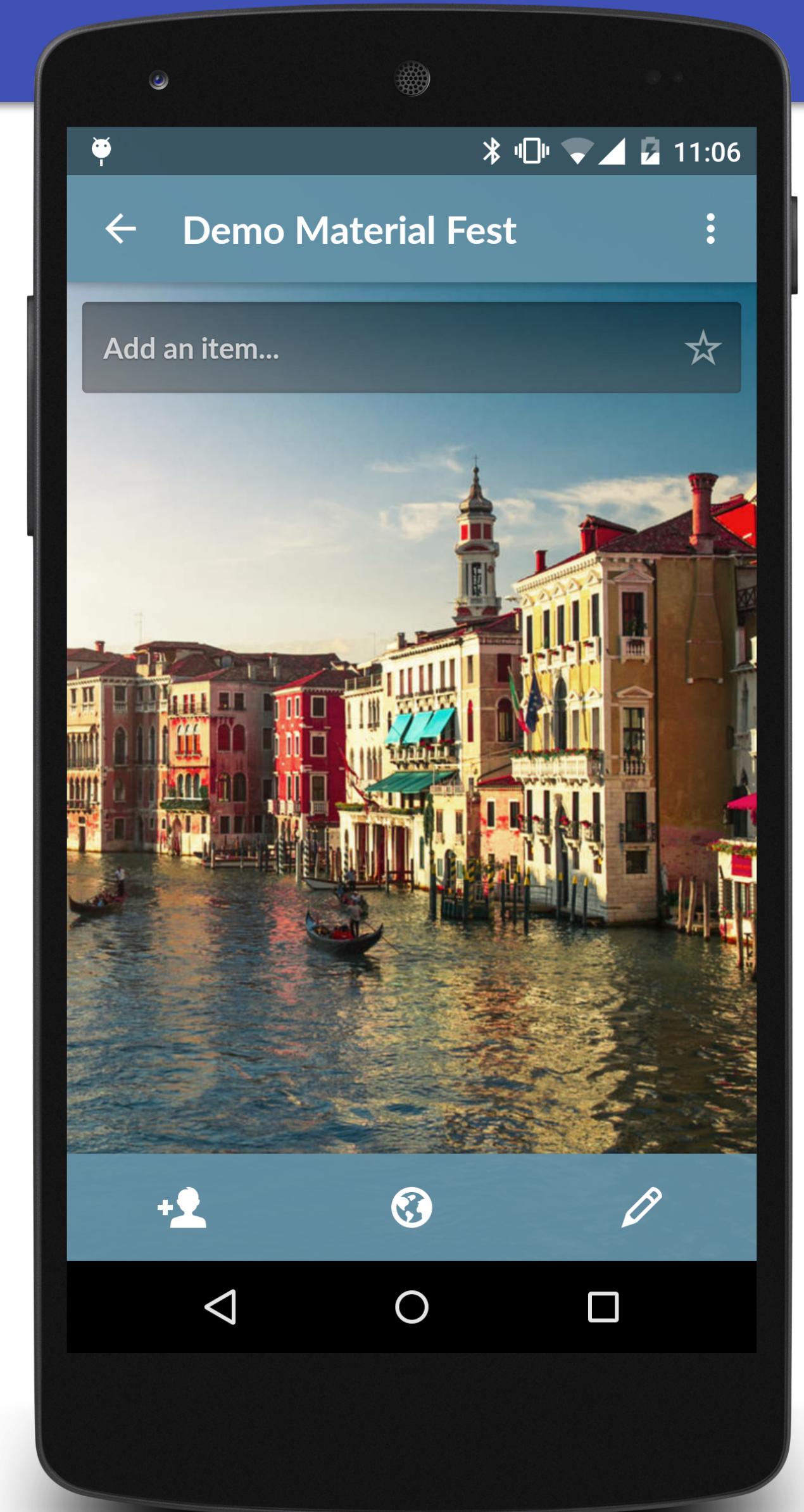
Tasks View - Toolbar

- Generalization of ActionBar.
- 48dip vs 56dip.
- Default elevation 4dip.
- More flexible, but more work needed.



Tasks View - Toolbar

- Generalization of ActionBar.
- 48dip vs 56dip.
- Default elevation 4dip.
- More flexible, but more work needed.



Tasks View - Toolbar

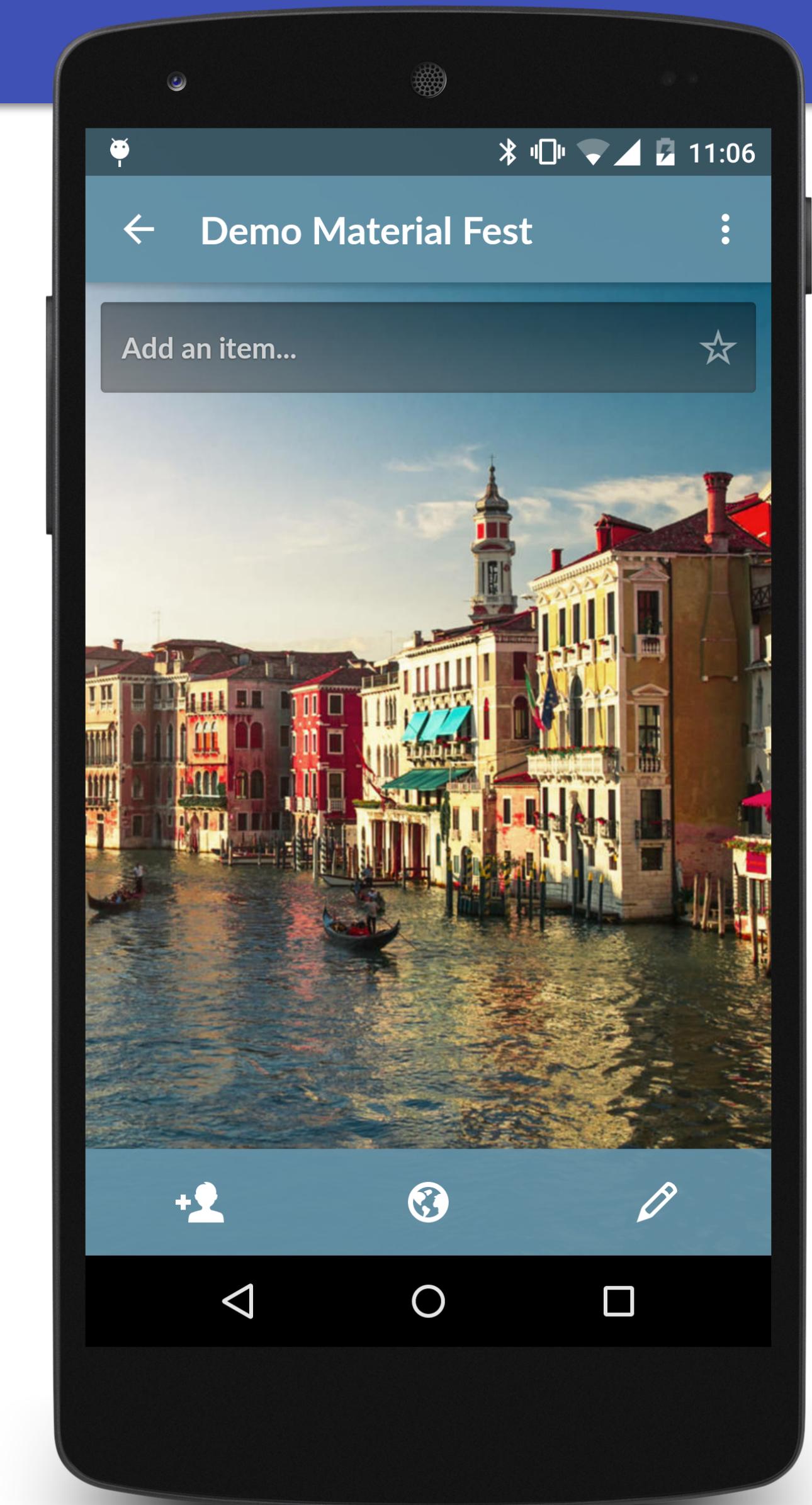
```
<android.support.v7.widget.Toolbar  
    app:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"  
    android:id="@+id/toolbar"  
    android:background="?attr/colorPrimary"  
    android:elevation="4dip">
```

```
<ProgressBar  
    android:id="@+id/progress_circular"  
    android:layout_gravity="right"  
    android:indeterminate="true"  
    android:visibility="gone"/>
```

```
</android.support.v7.widget.Toolbar>
```

```
<LinearLayout>  
    <include layout="@layout/toolbar"/>  
</LinearLayout>
```

```
Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
setSupportActionBar(toolbar);
```



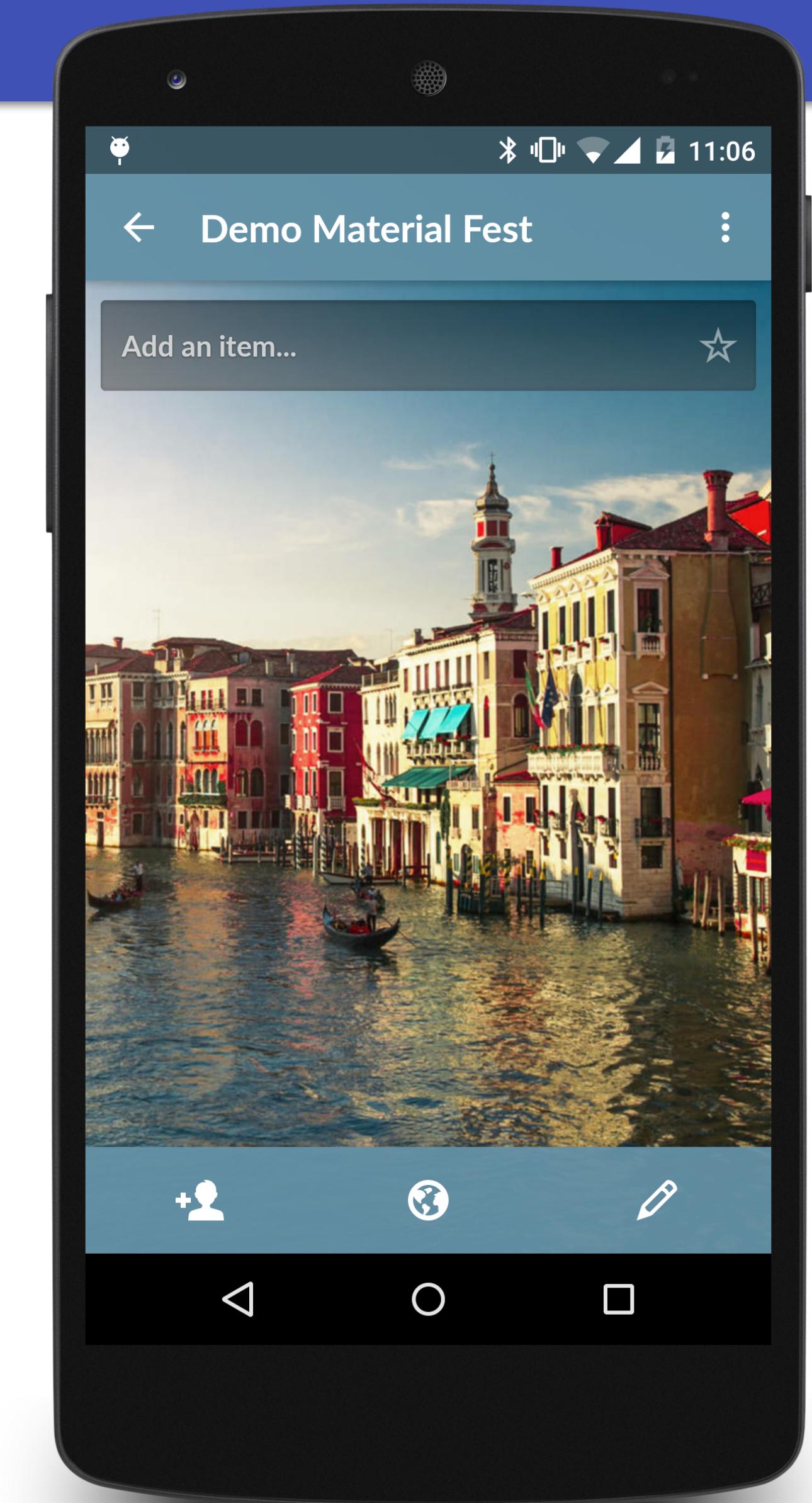
Tasks View - Toolbar

```
<android.support.v7.widget.Toolbar  
    app:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"  
    android:id="@+id/toolbar"  
    android:background="?attr/colorPrimary"  
    android:elevation="4dip">
```

```
<ProgressBar  
    android:id="@+id/progress_circular"  
    android:layout_gravity="right"  
    android:indeterminate="true"  
    android:visibility="gone"/>  
</android.support.v7.widget.Toolbar>
```

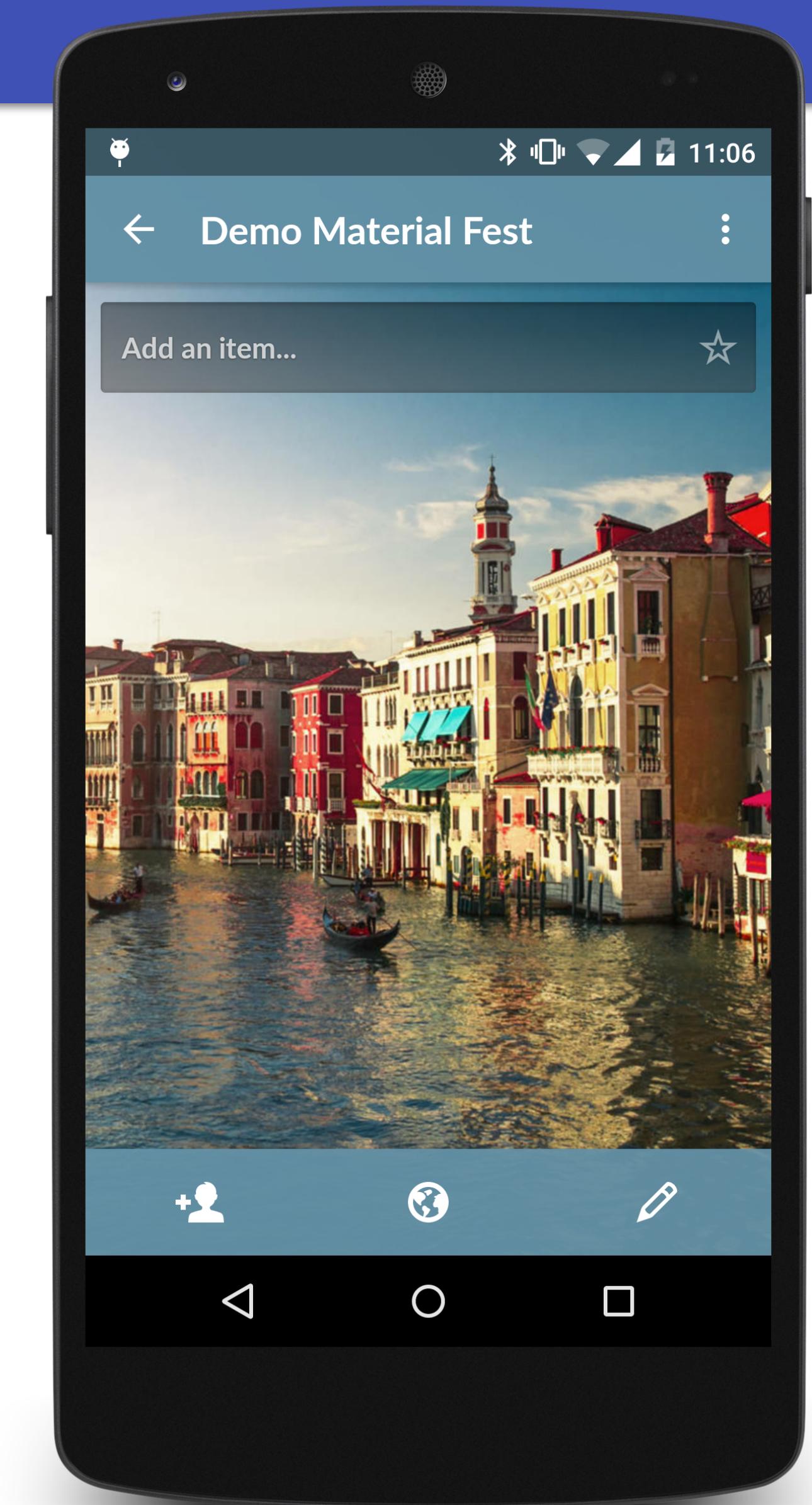
```
<LinearLayout>  
    <include layout="@layout/toolbar"/>  
</LinearLayout>
```

```
Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
setSupportActionBar(toolbar);
```



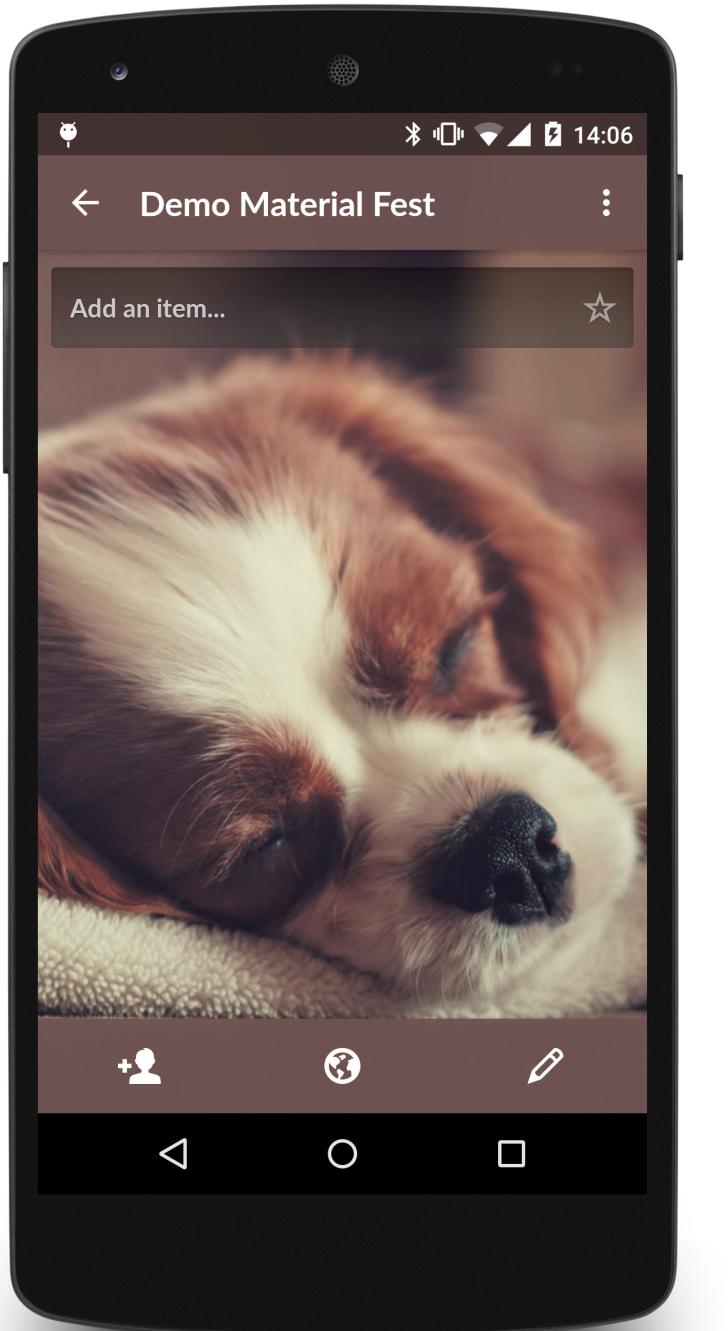
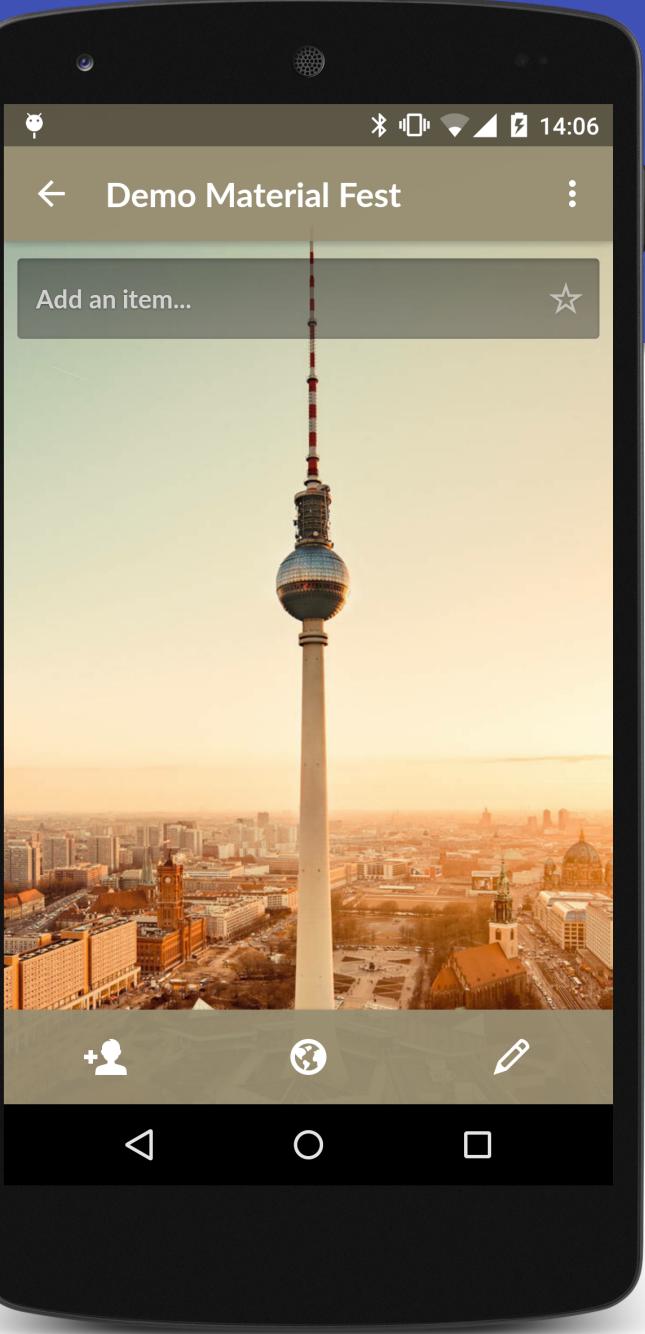
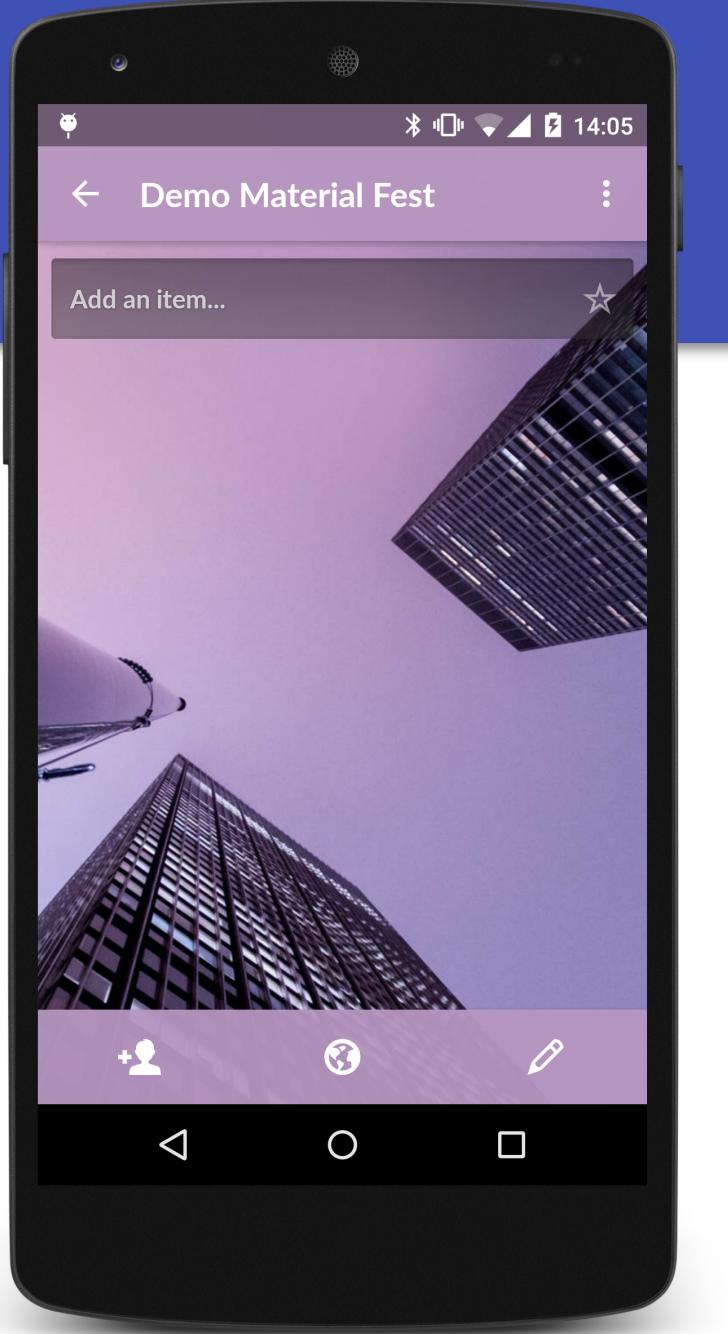
Tasks View - Toolbar

```
<android.support.v7.widget.Toolbar  
    app:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"  
    android:id="@+id/toolbar"  
    android:background="?attr/colorPrimary"  
    android:elevation="4dip">  
  
<ProgressBar  
    android:id="@+id/progress_circular"  
    android:layout_gravity="right"  
    android:indeterminate="true"  
    android:visibility="gone"/>  
</android.support.v7.widget.Toolbar>  
  
<LinearLayout>  
    <include layout="@layout/toolbar"/>  
</LinearLayout>  
  
Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
setSupportActionBar(toolbar);
```



Tasks View - Palette

- Not part of SDK, in support library.
- Extract colours from an image.
- Personalize UI components.
- It tries to generate 6 swatches.



Tasks View - Palette

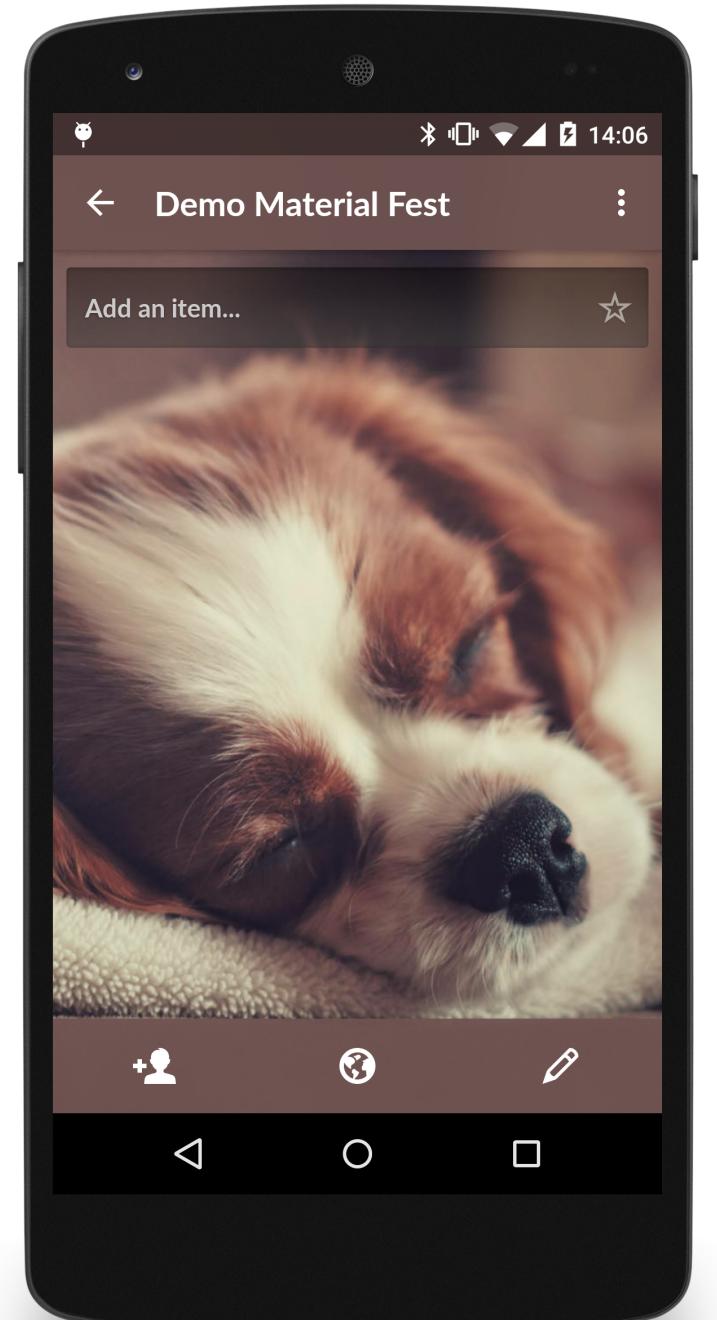
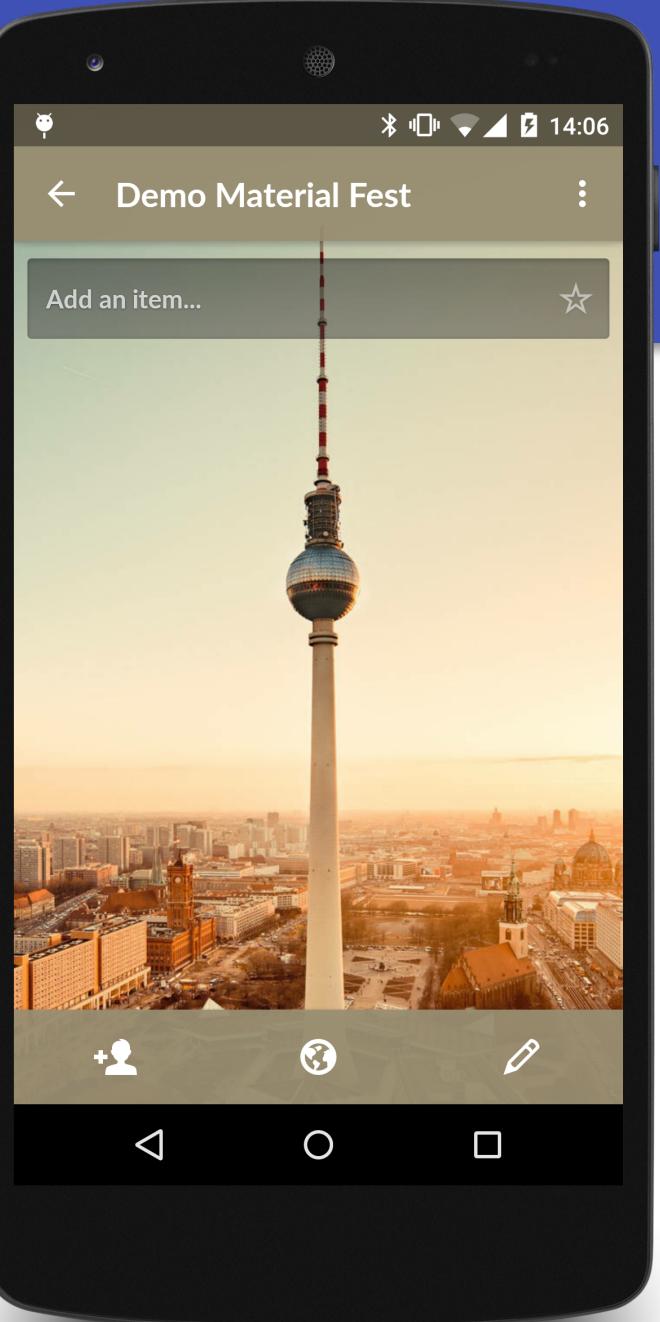
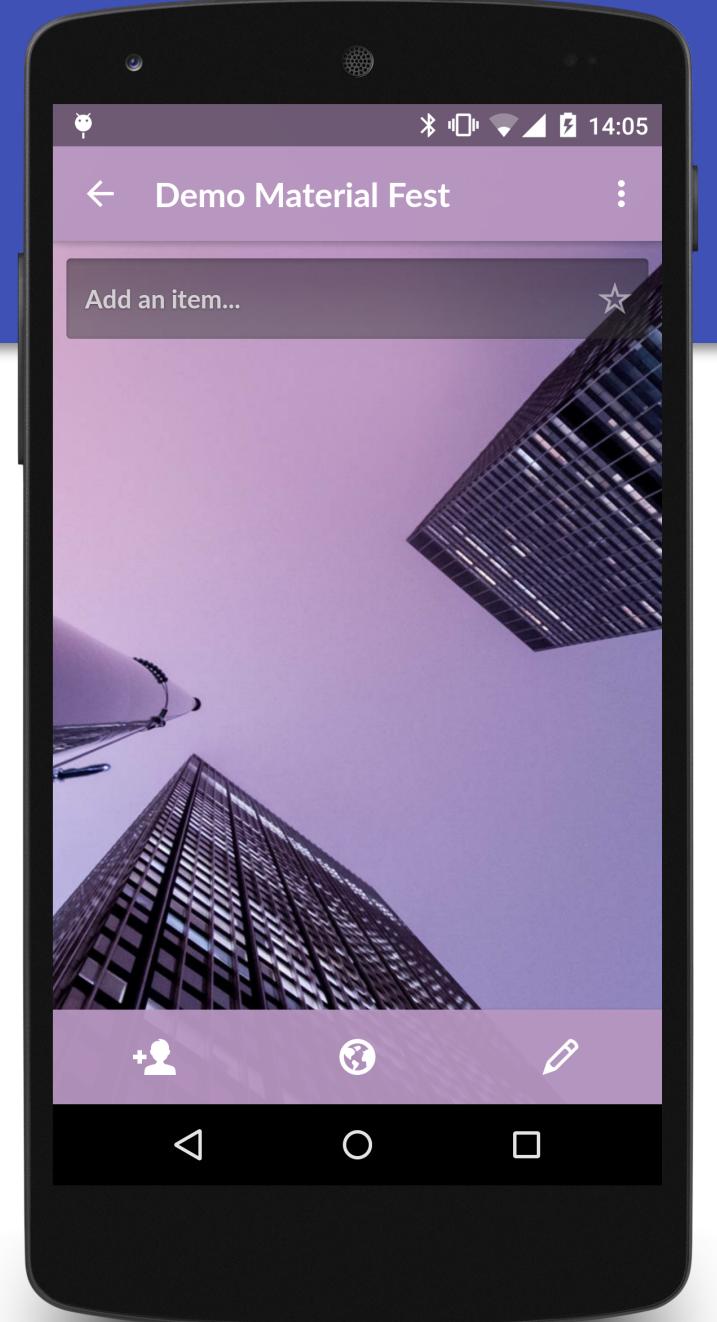
```
compile 'com.android.support:palette-v7:21.0.3'
```

```
Palette.generateAsync(bitmap, new PaletteAsyncListener() {
```

```
    @Override  
    public void onGenerated(Palette palette) {  
        if (palette != null) {
```

```
            ColorDrawable color = new ColorDrawable(  
                palette.getMutedColor(defaultColor)  
            );  
            color.setAlpha(200);  
            getSupportActionBar().setBackgroundDrawable(color);
```

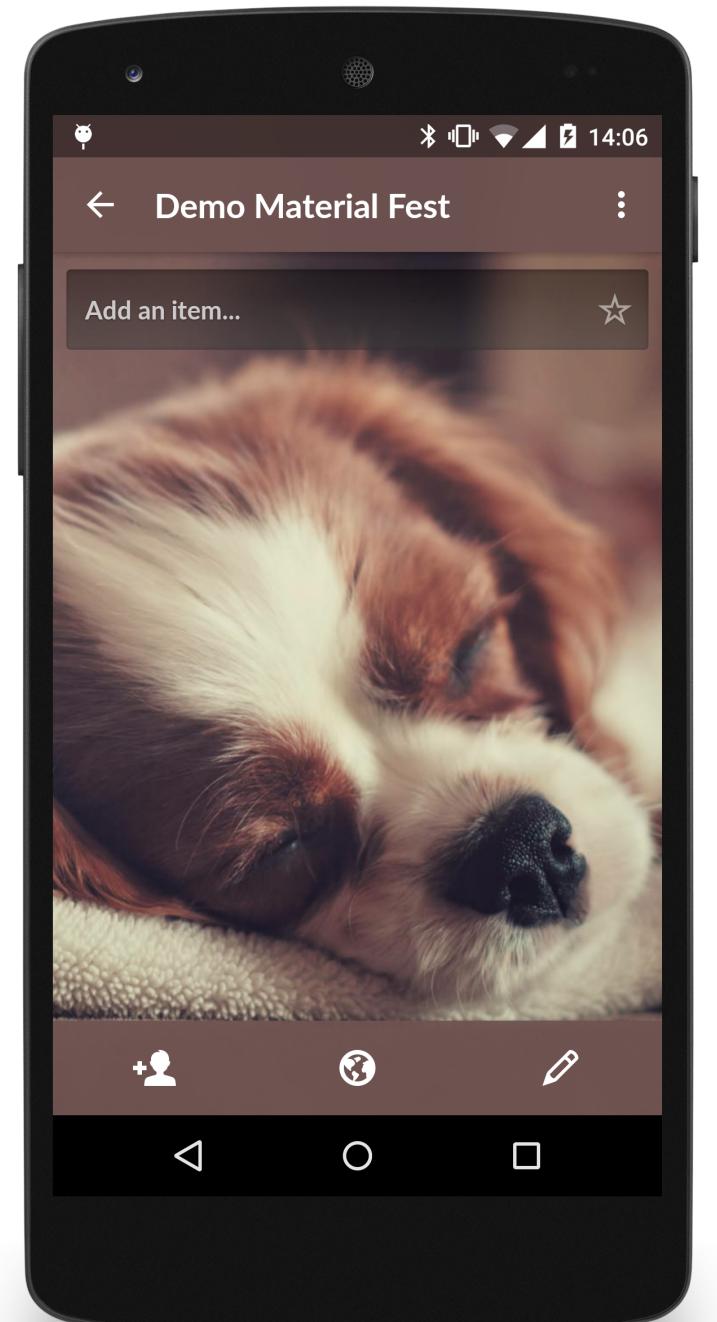
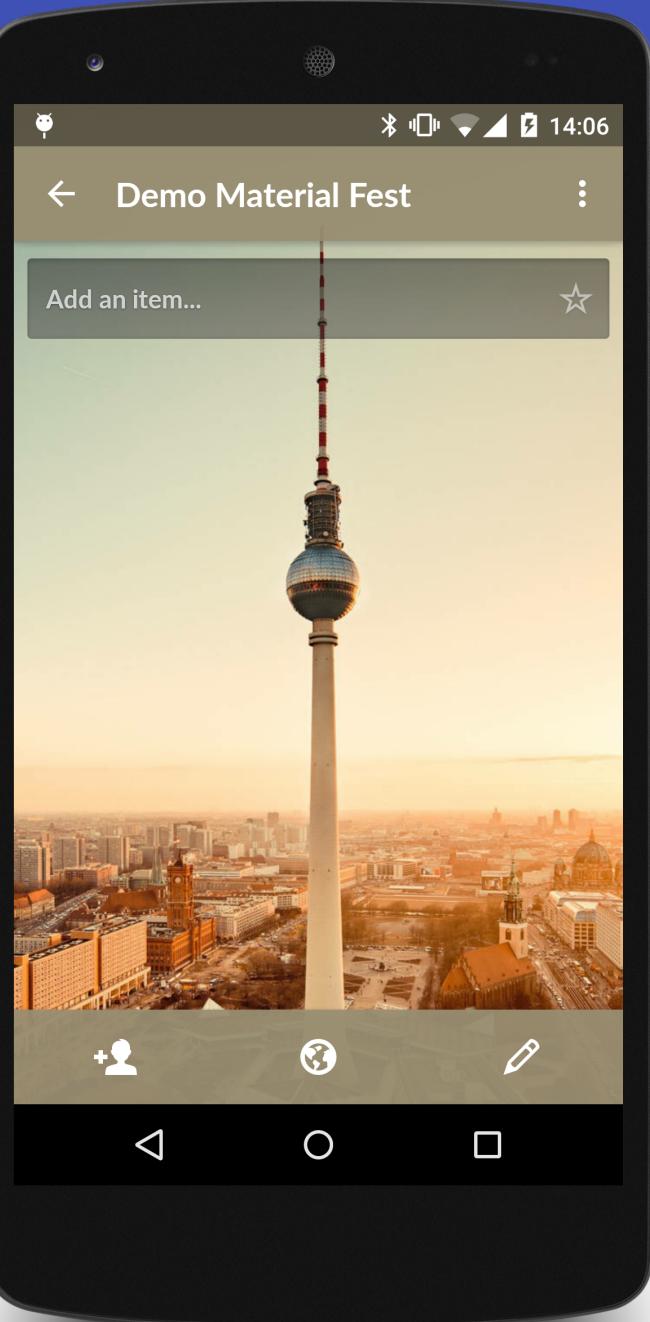
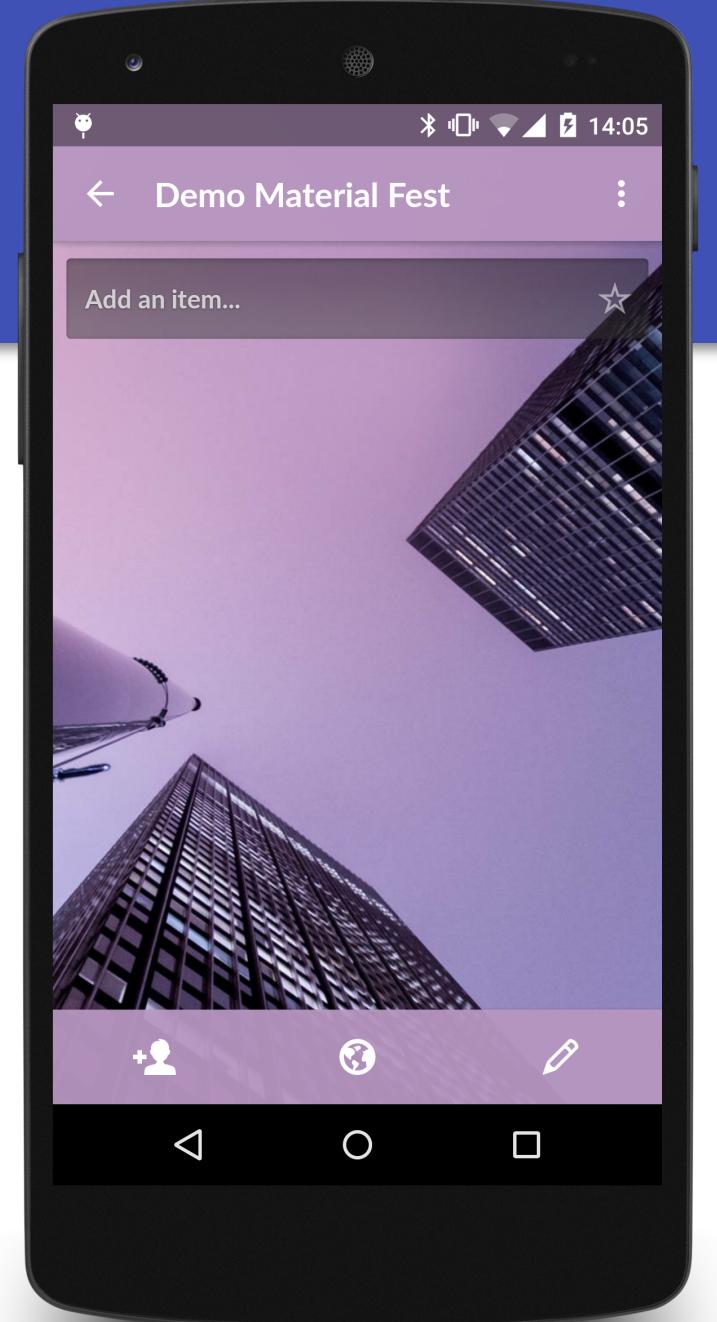
```
    }  
});
```



Tasks View - Palette

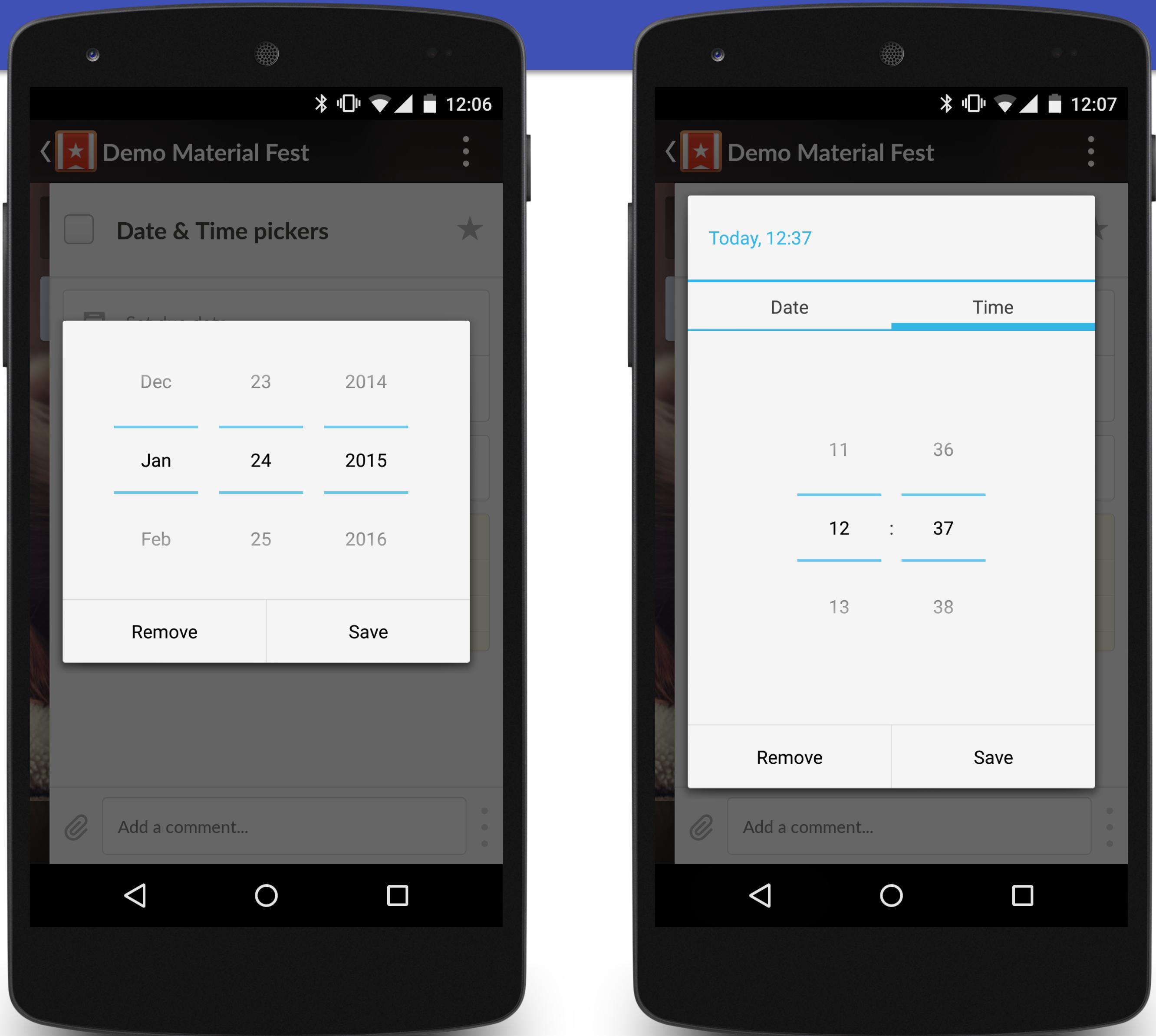
```
compile 'com.android.support:palette-v7:21.0.3'
```

```
Palette.generateAsync(bitmap, new PaletteAsyncListener() {  
  
    @Override  
    public void onGenerated(Palette palette) {  
        if (palette != null) {  
  
            ColorDrawable color = new ColorDrawable(  
                palette.getMutedColor(defaultColor)  
            );  
            color.setAlpha(200);  
            getSupportActionBar().setBackgroundDrawable(color);  
  
        }  
    }  
});
```



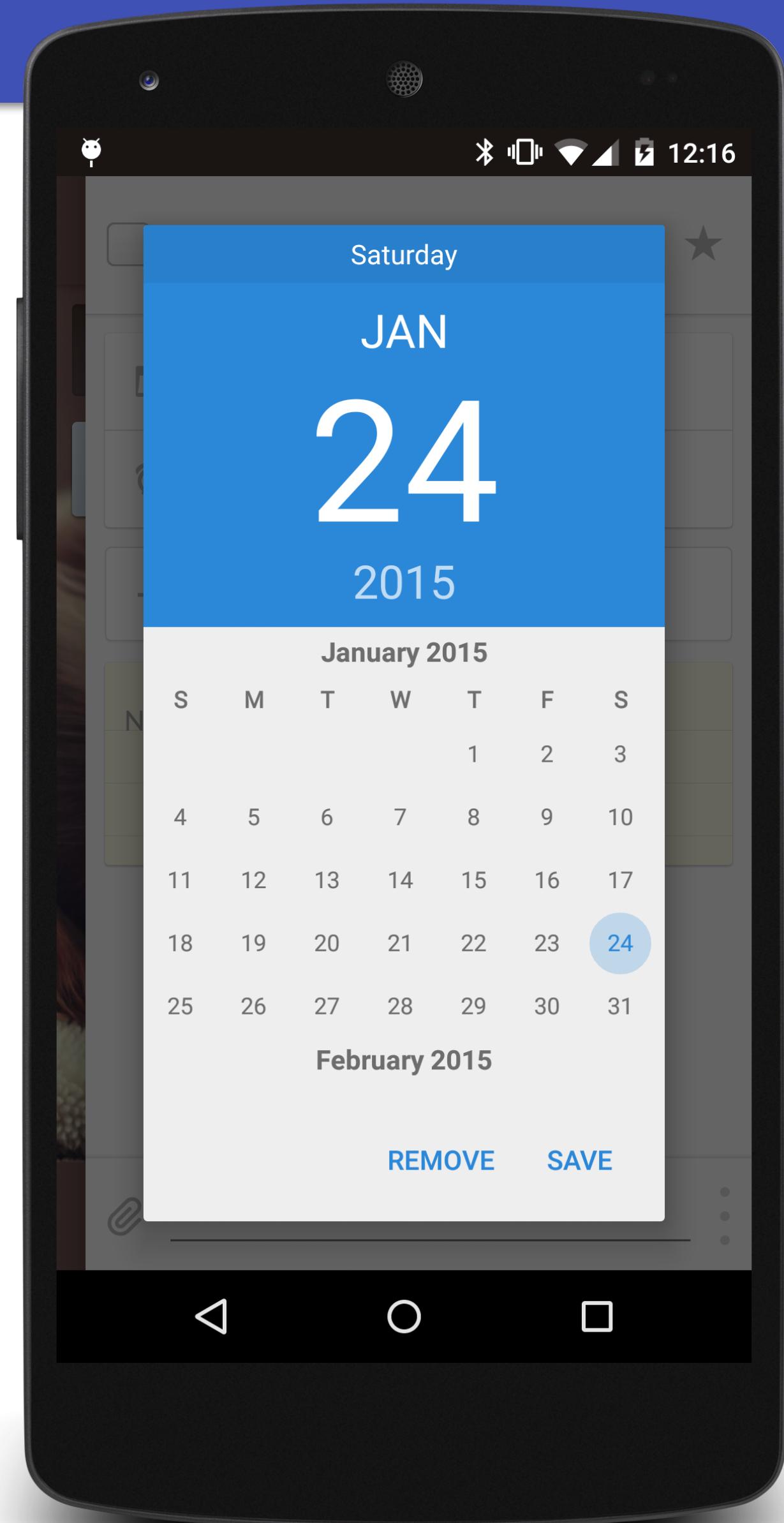
Details View - Pickers

- Old-fashioned.
- Slow to select date/time.
- Almost “for free”.
- Use colorAccent.



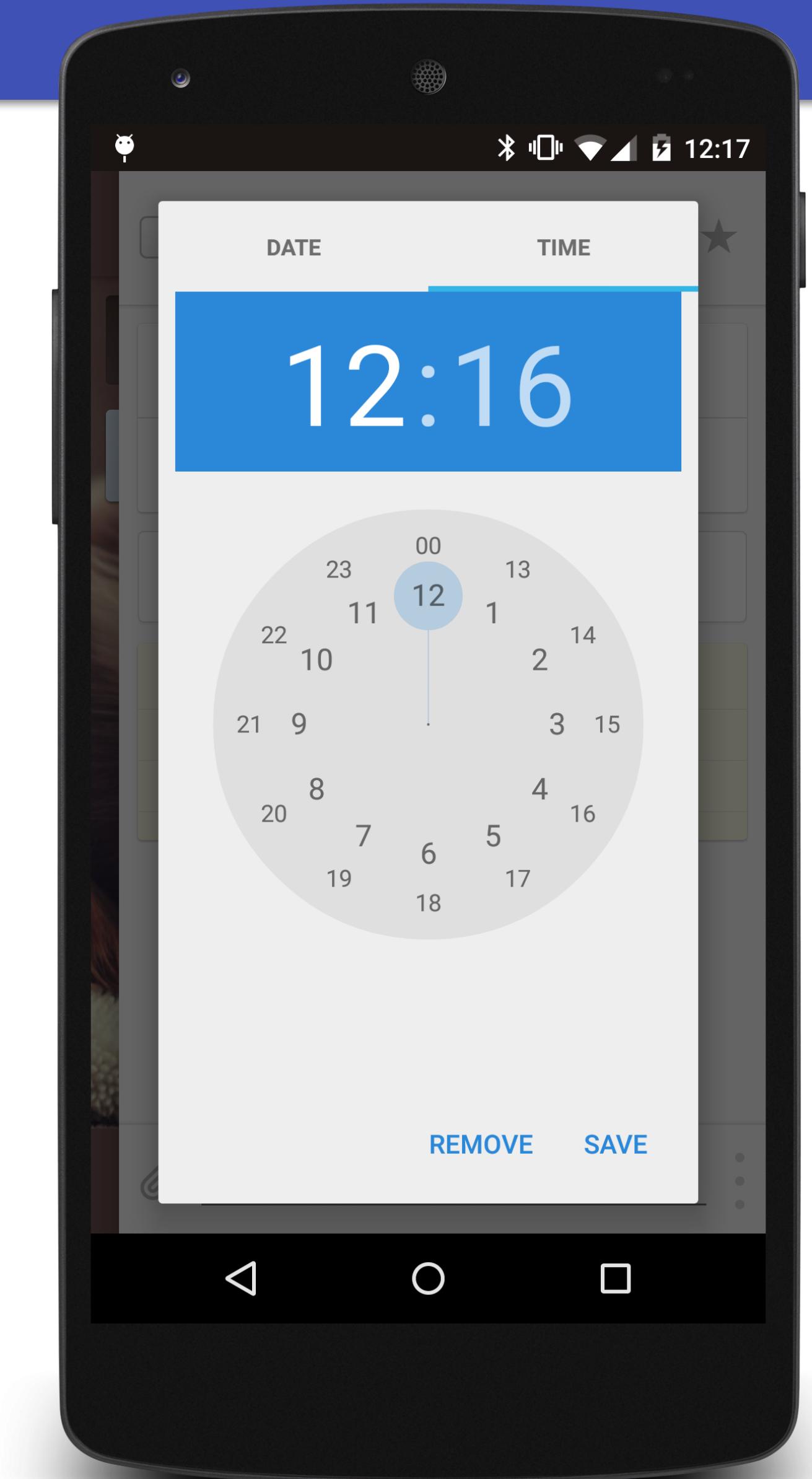
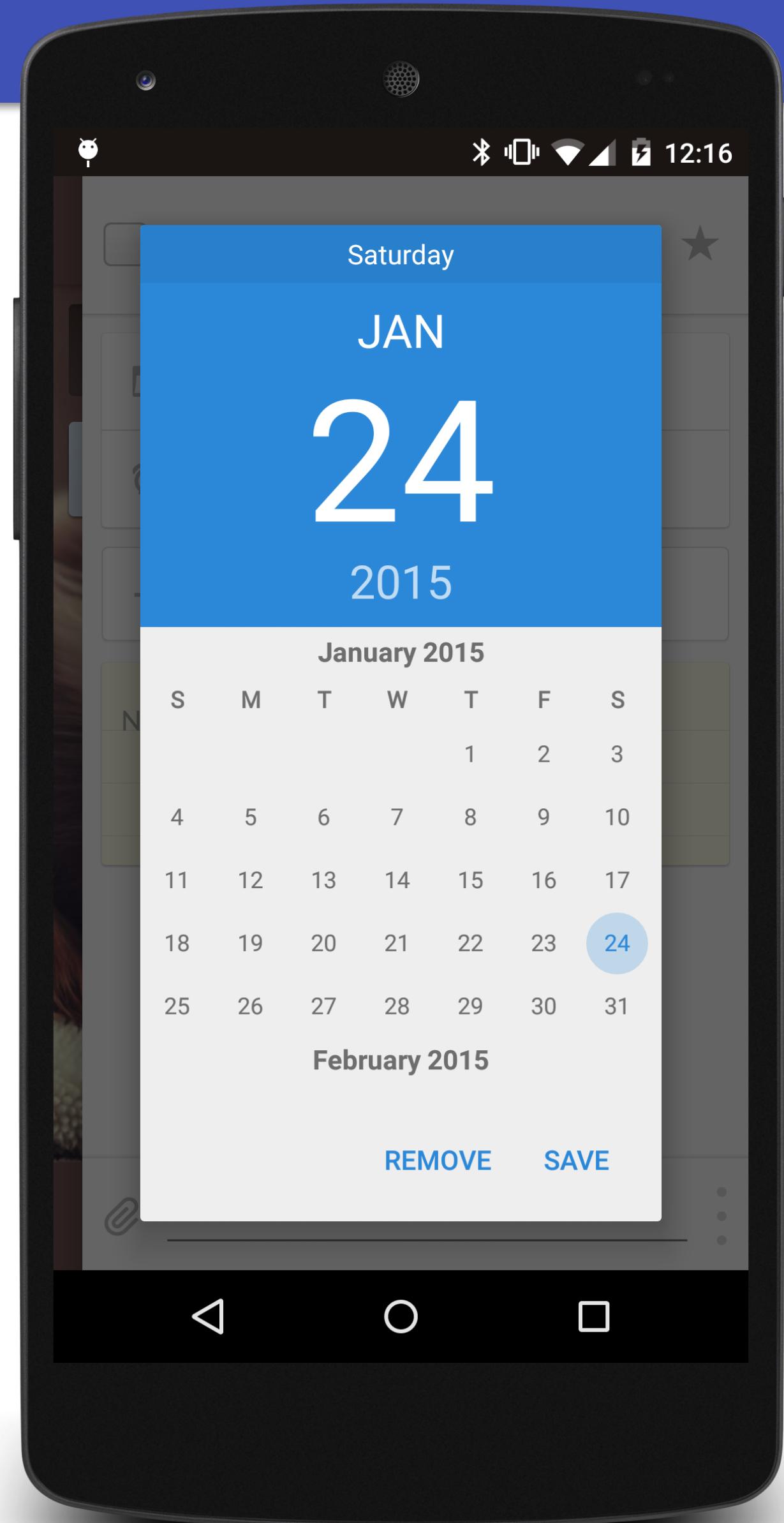
Details View - Pickers

- Old-fashioned.
- Slow to select date/time.
- Almost “for free”.
- Use colorAccent.



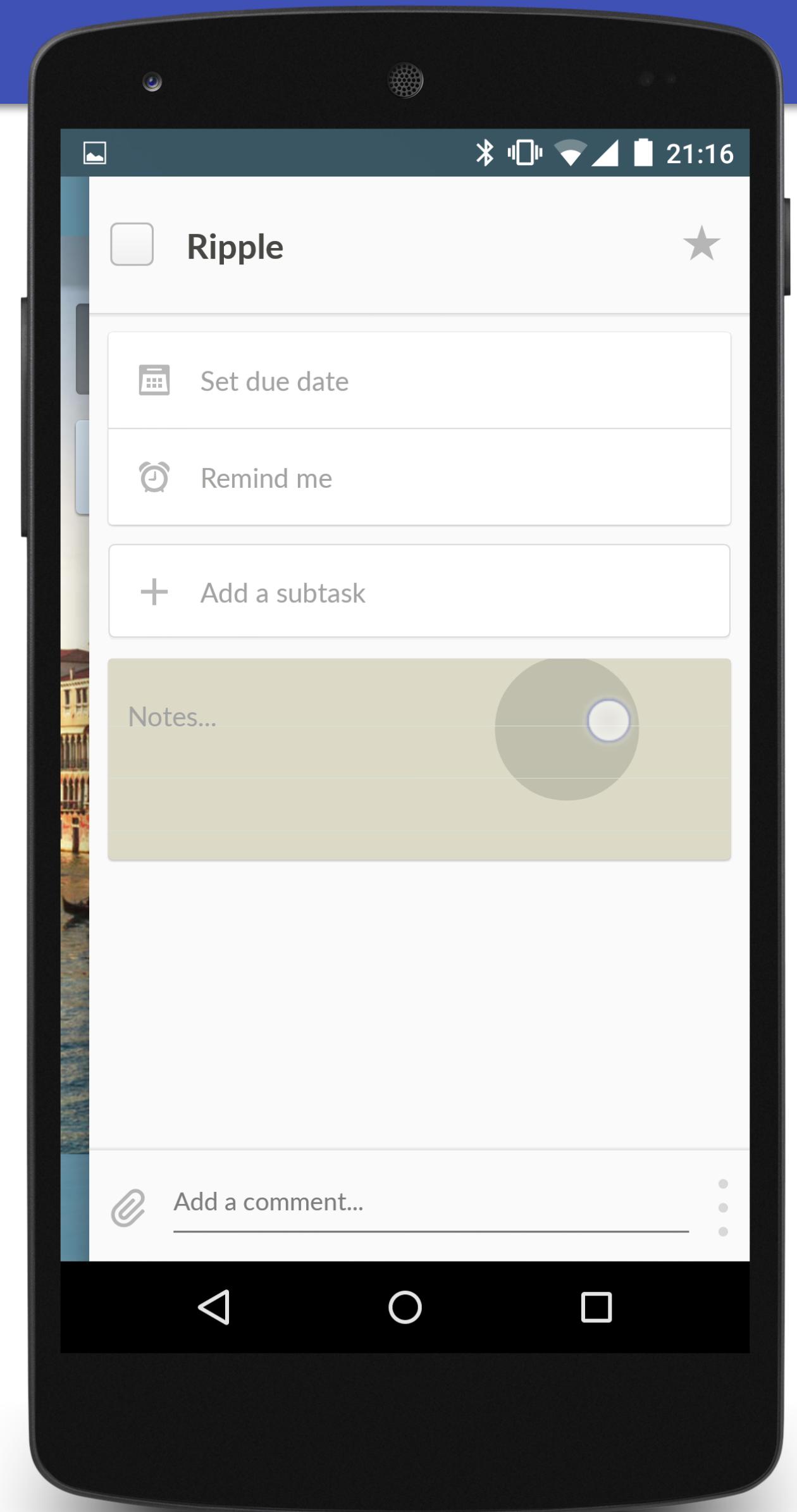
Details View - Pickers

- Old-fashioned.
- Slow to select date/time.
- Almost “for free”.
- Use colorAccent.



Details View - Ripple

- One of key features of Material Design.
- Tangible surfaces.
- selectableItemBackground (Borderless).
- RippleDrawable.



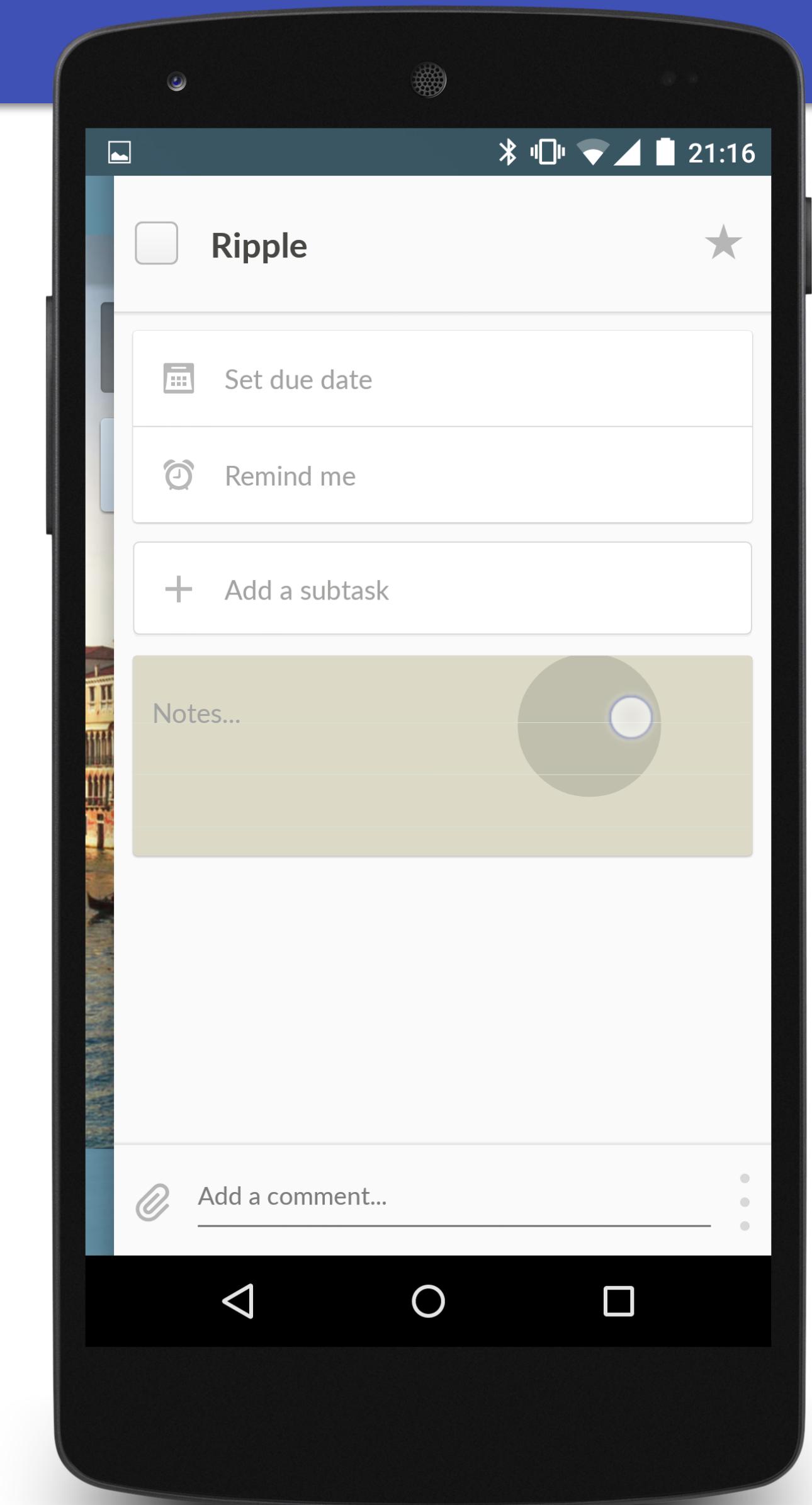
Details View - Ripple

```
<style name="WunderlistTheme.Details">
    <item name="colorControlHighlight">@color/gray</item>
</style>

<FrameLayout
    android:background="@color/yellow"
    android:foreground="?attr/selectableItemBackground"/>

<ripple
    android:color="?attr/colorControlHighlight">
    <item android:drawable="@color/yellow" />
</ripple>

<FrameLayout
    android:background="@drawable/ripple"/>
```



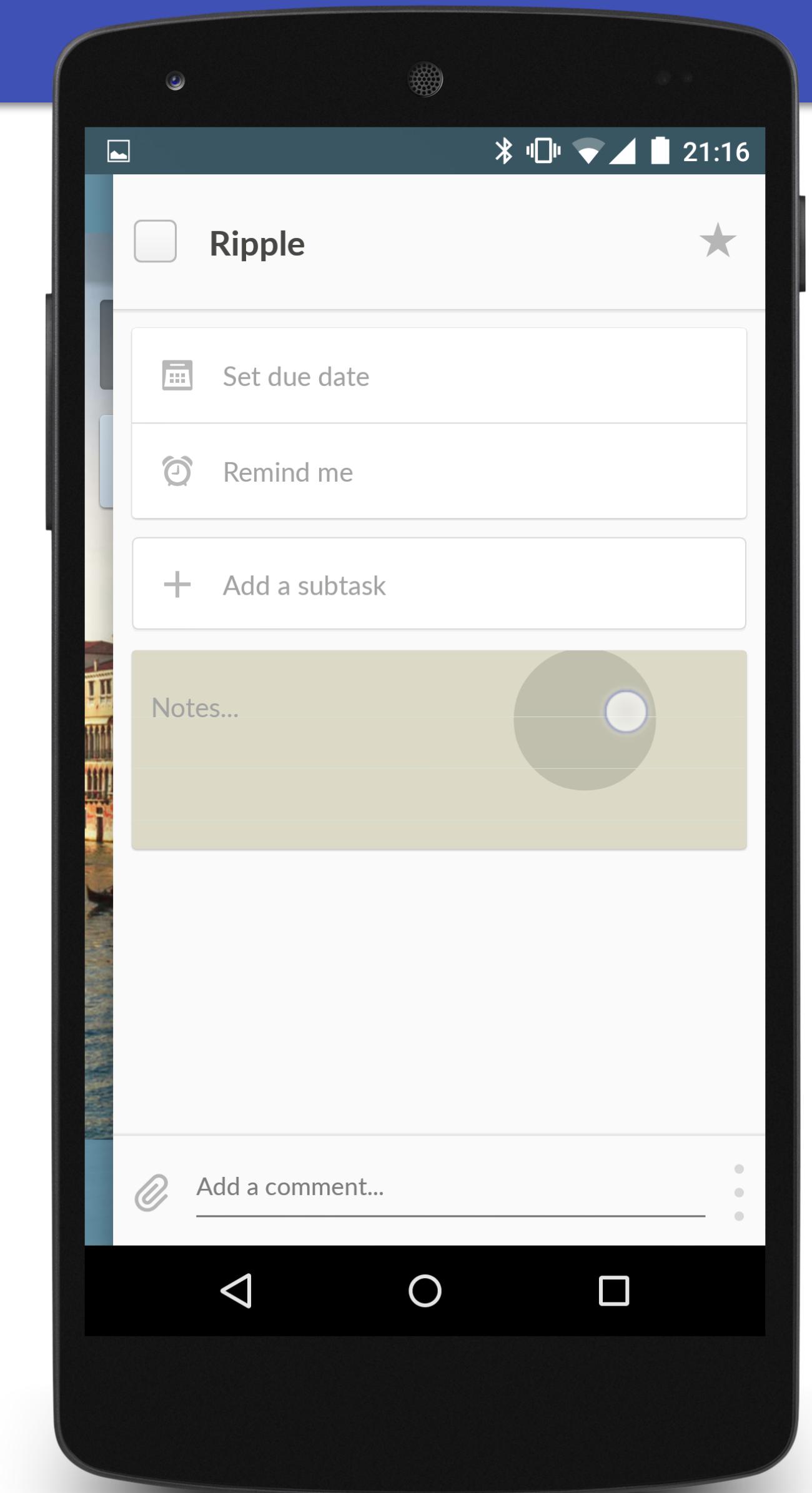
Details View - Ripple

```
<style name="WunderlistTheme.Details">
    <item name="colorControlHighlight">@color/gray</item>
</style>

<FrameLayout
    android:background="@color/yellow"
    android:foreground="?attr/selectableItemBackground"/>

<ripple
    android:color="?attr/colorControlHighlight">
    <item android:drawable="@color/yellow" />
</ripple>

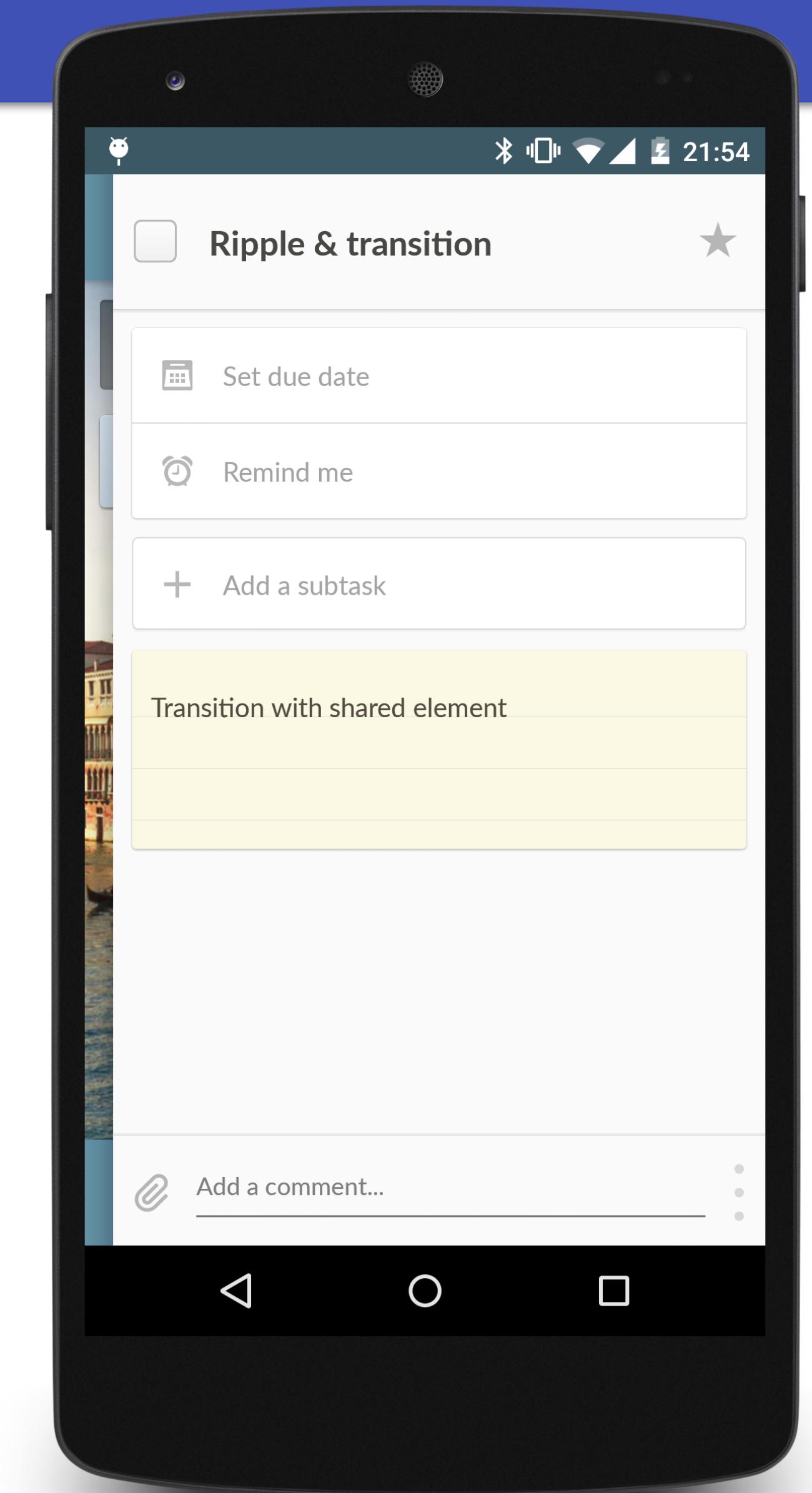
<FrameLayout
    android:background="@drawable/ripple"/>
```



Details View - Note transition

- Meaningful motion.
- Share views between activities.
- API 21+ only.

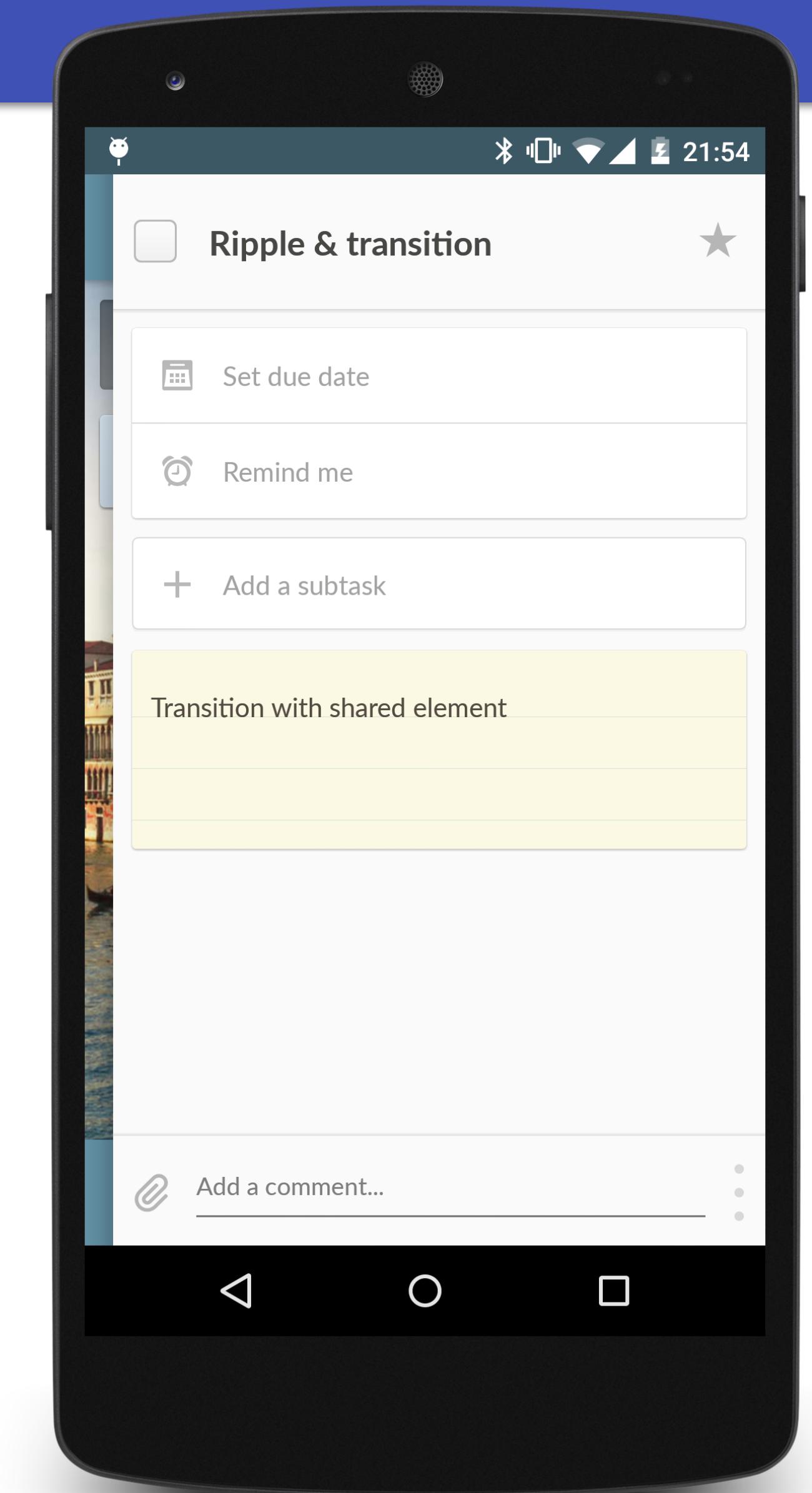
```
<style name="WunderlistTheme" parent="WunderlistTheme.Base">
    <item name="android:windowContentTransitions">true</item>
    <item name="android:windowSharedElementEnterTransition">
        @android:transition/move
    </item>
</style>
```



Details View - Note transition

- Meaningful motion.
- Share views between activities.
- API 21+ only.

```
<style name="WunderlistTheme" parent="WunderlistTheme.Base">
    <item name="android:windowContentTransitions">true</item>
    <item name="android:windowSharedElementEnterTransition">
        @android:transition/move
    </item>
</style>
```



Details View - Note transition

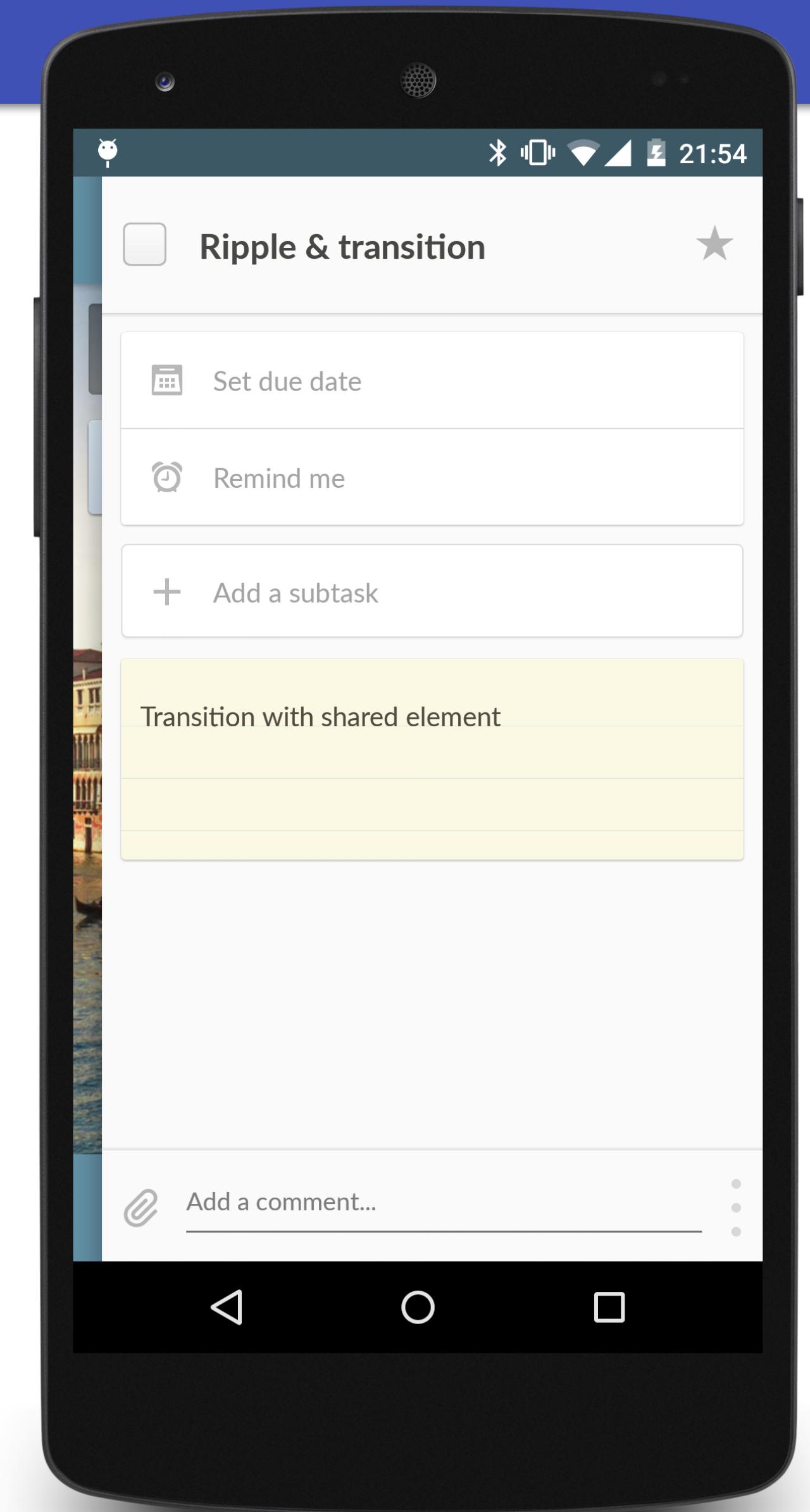
<EditText

```
    android:transitionName="Note transition"/>
```

```
Intent intent = new Intent(context, NoteActivity.class);
intent.putExtra(EXTRA_TASK_NOTE, note);
```

```
ActivityOptionsCompat options =
    ActivityOptionsCompat.makeSceneTransitionAnimation(
        context,
        noteTextView,
        "Note transition"
    );
```

```
ActivityCompat.startActivityForResult(context, intent,
    REQUEST_CODE, options.toBundle());
```



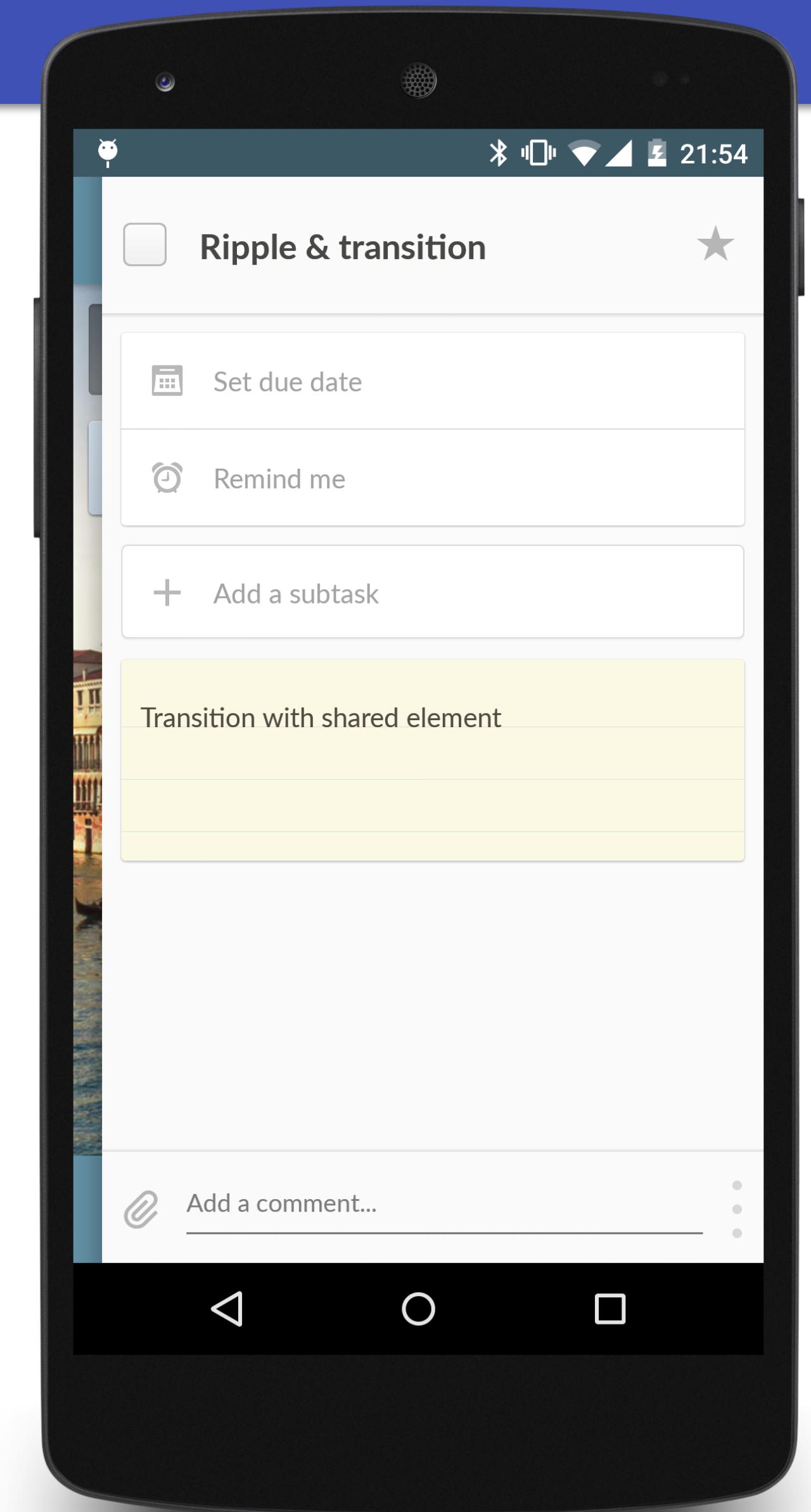
Details View - Note transition

```
<EditText  
    android:transitionName="Note transition"/>
```

```
Intent intent = new Intent(context, NoteActivity.class);  
intent.putExtra(EXTRA_TASK_NOTE, note);
```

```
ActivityOptionsCompat options =  
    ActivityOptionsCompat.makeSceneTransitionAnimation(  
        context,  
        noteTextView,  
        "Note transition"  
    );
```

```
ActivityCompat.startActivityForResult(context, intent,  
    REQUEST_CODE, options.toBundle());
```



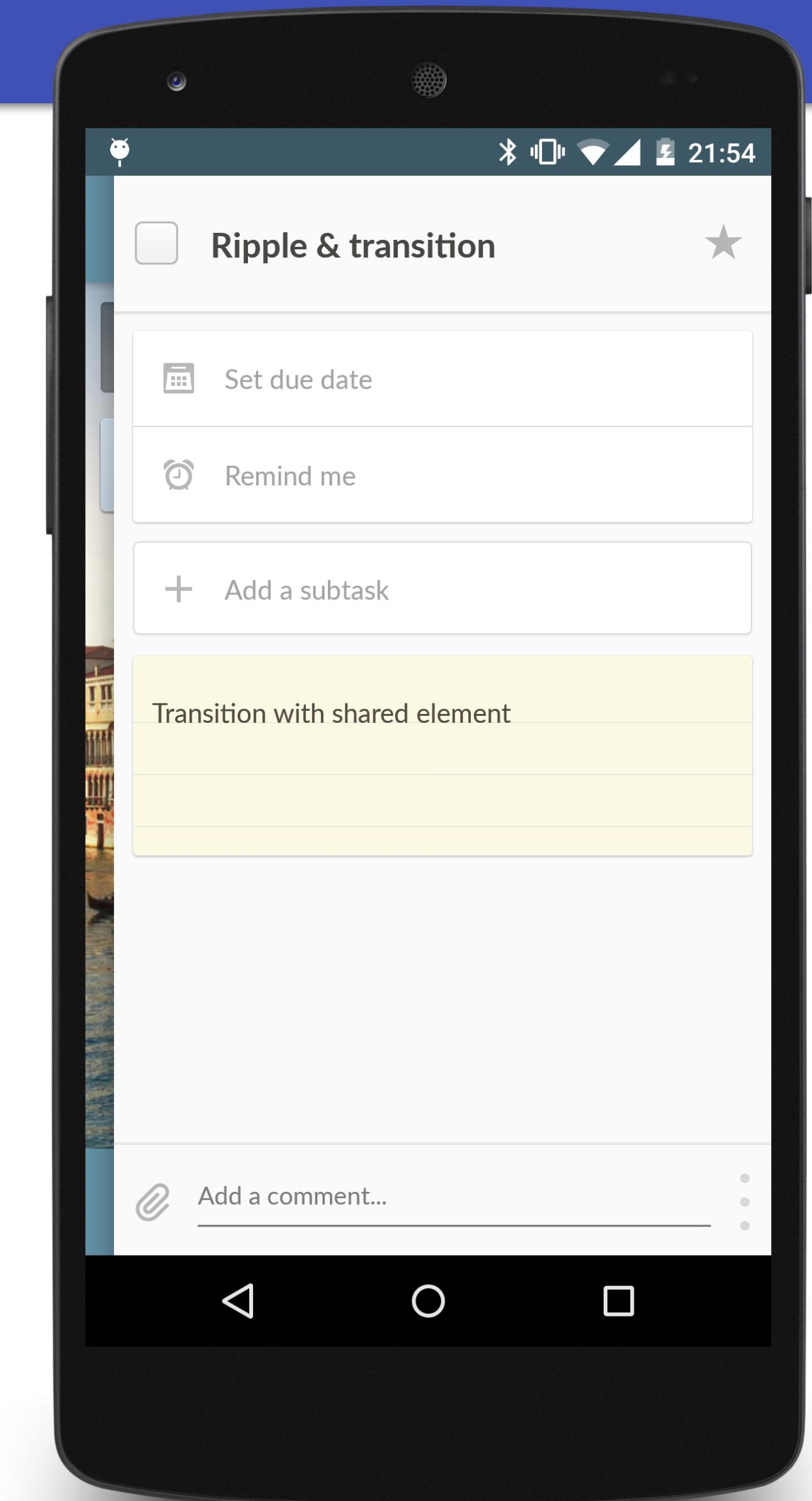
Details View - Note transition

```
<EditText  
    android:transitionName="Note transition"/>
```

```
Intent intent = new Intent(context, NoteActivity.class);  
intent.putExtra(EXTRA_TASK_NOTE, note);
```

```
ActivityOptionsCompat options =  
    ActivityOptionsCompat.makeSceneTransitionAnimation(  
        context,  
        noteTextView,  
        "Note transition"  
    );
```

```
ActivityCompat.startActivityForResult(context, intent,  
    REQUEST_CODE, options.toBundle());
```



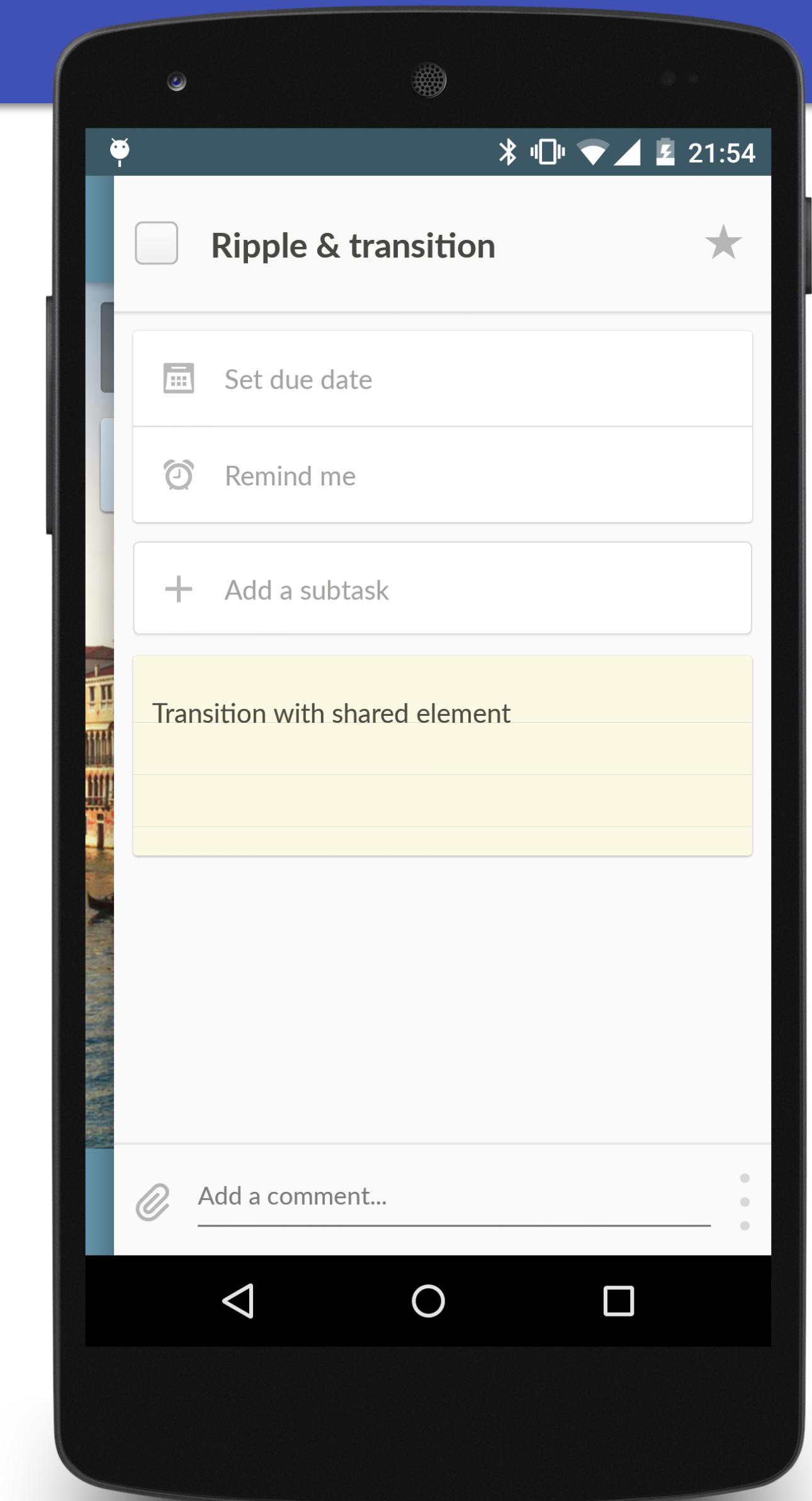
Details View - Note transition

```
<EditText  
    android:transitionName="Note transition"/>
```

```
Intent intent = new Intent(context, NoteActivity.class);  
intent.putExtra(EXTRA_TASK_NOTE, note);
```

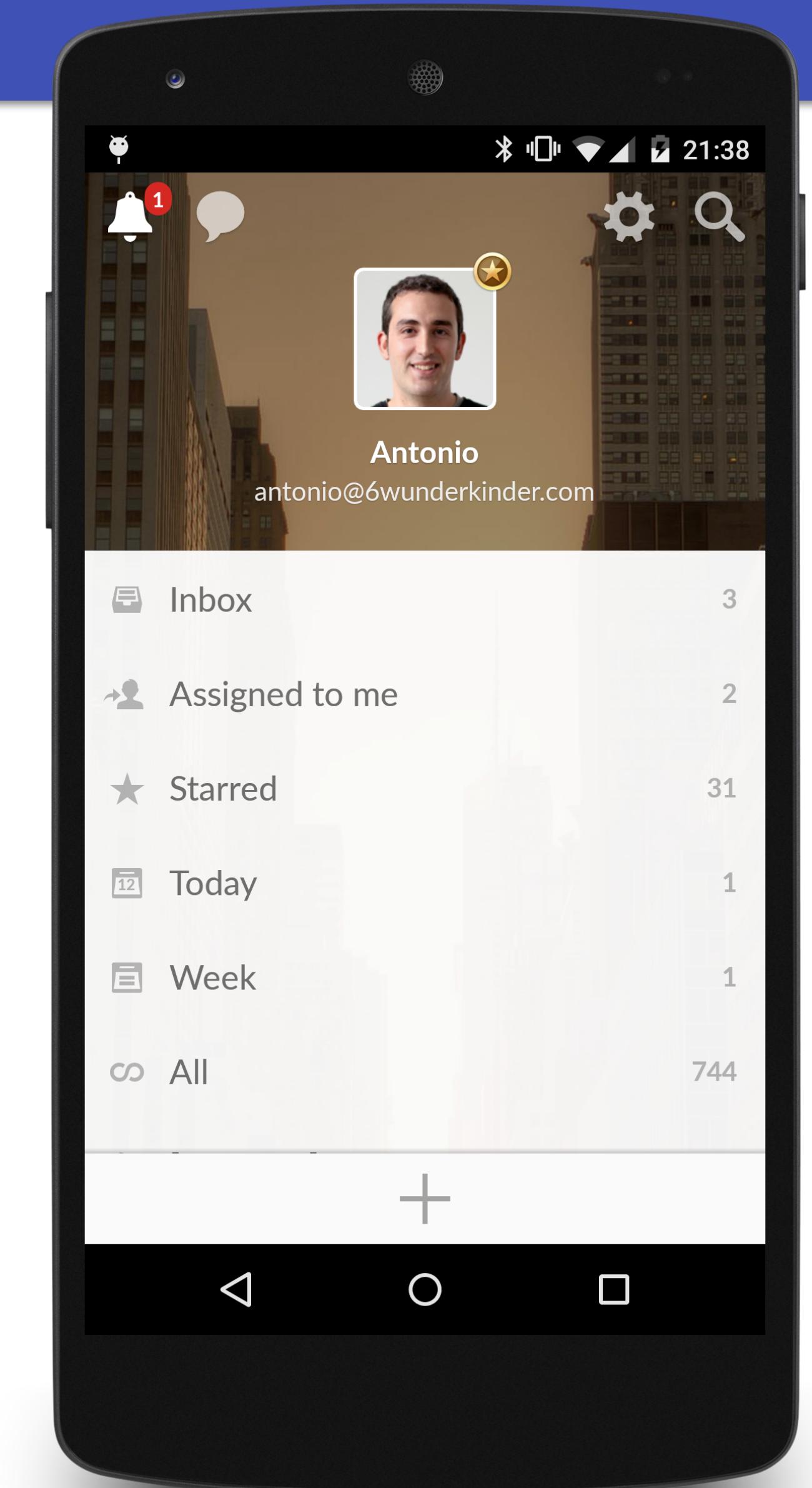
```
ActivityOptionsCompat options =  
    ActivityOptionsCompat.makeSceneTransitionAnimation(  
        context,  
        noteTextView,  
        "Note transition"  
    );
```

```
ActivityCompat.startActivityForResult(context, intent,  
    REQUEST_CODE, options.toBundle());
```



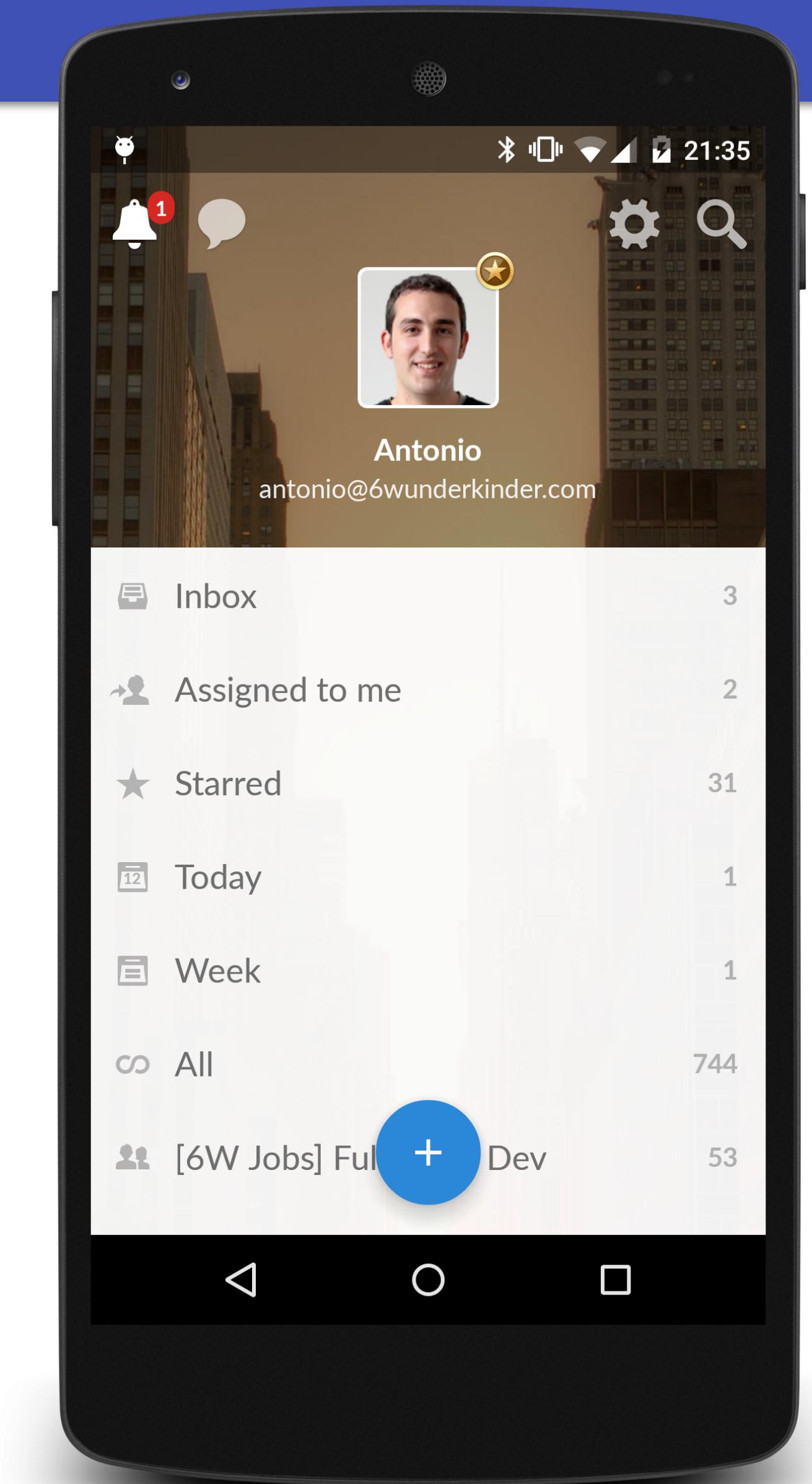
Home View - FAB and Nav. drawer?

- Circular button made of paper.
- Lifts and emits ink reactions on press.
- Used for promoted actions, not mandatory.
- Never use it for destructive actions.



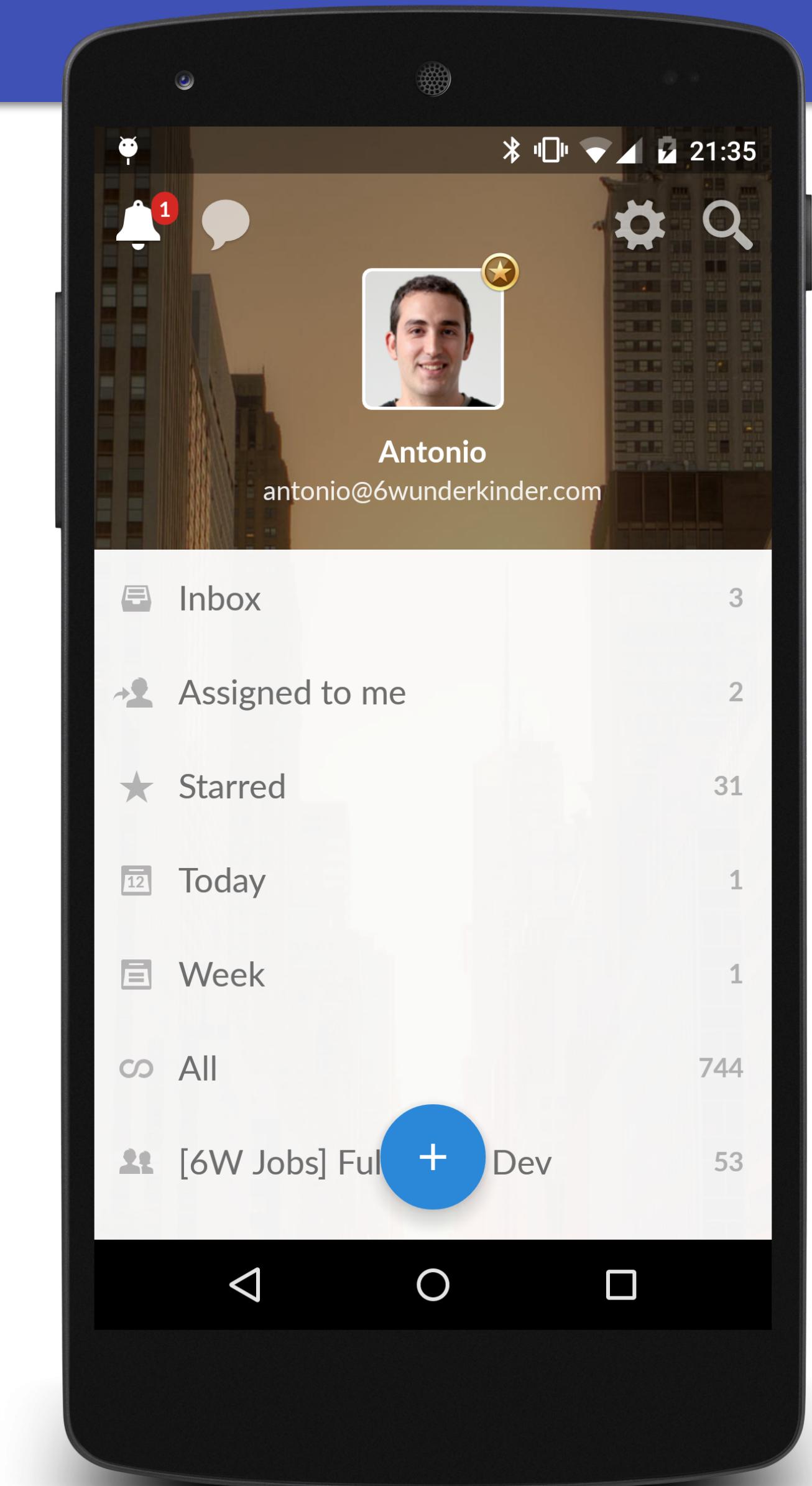
Home View - FAB and Nav. drawer?

- Circular button made of paper.
- Lifts and emits ink reactions on press.
- Used for promoted actions, not mandatory.
- Never use it for destructive actions.



Home View - FAB and Nav. drawer?

- Circular button made of paper.
- Lifts and emits ink reactions on press.
- Used for promoted actions, not mandatory.
- Never use it for destructive actions.

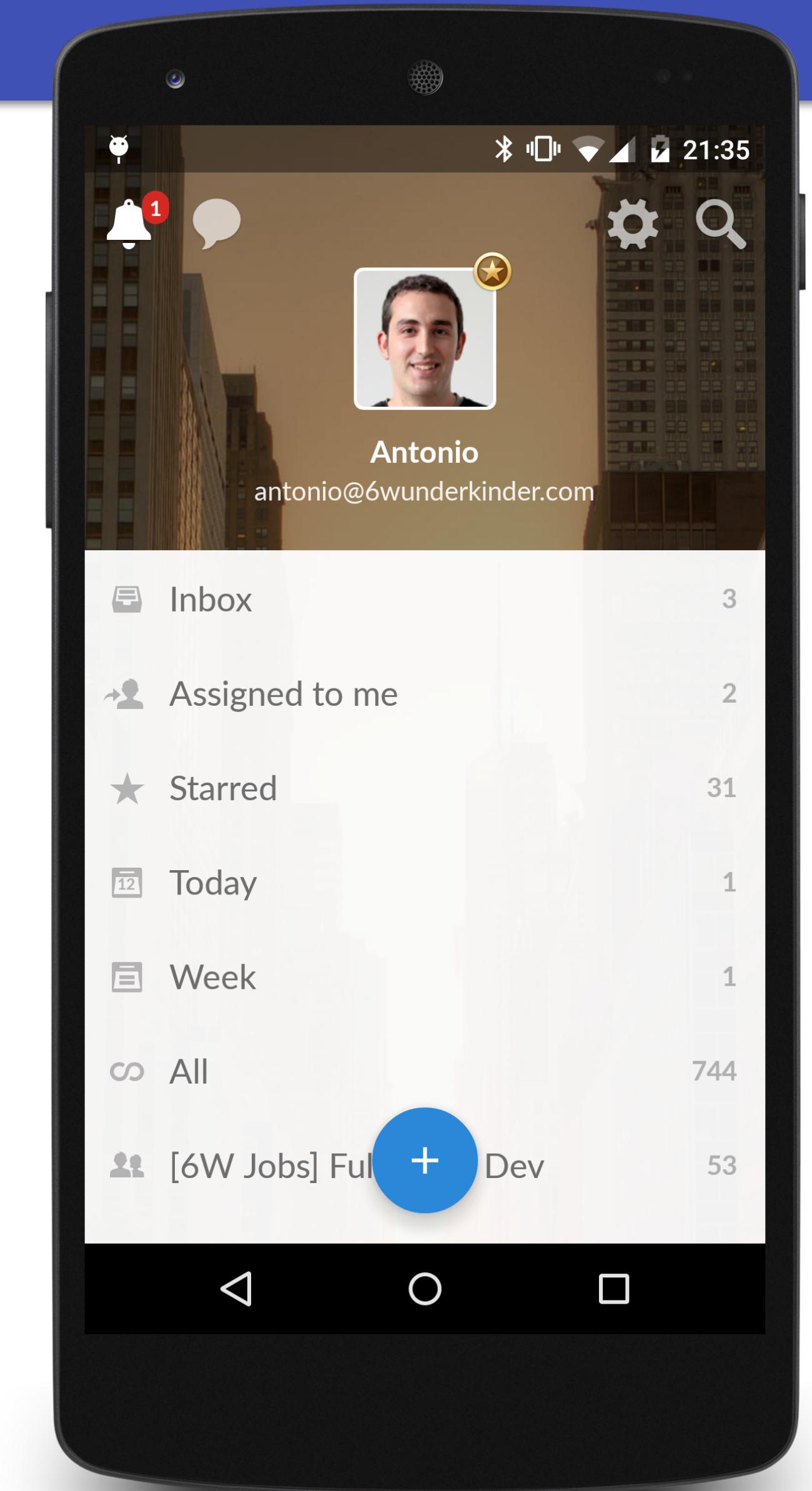


Home View - FAB

```
<ripple android:color="?attr/colorControlHighlight">
    <item android:drawable="@drawable/shape_fab_list_button"/>
</ripple>

<ImageButton
    android:layout_width="?attr/actionBarSize"
    android:layout_height="?attr/actionBarSize"
    android:layout_marginBottom="16dip"
    android:background="@drawable/fab_list_background"
    android:src="@drawable/ic_add"
    android:tint="@android:color/white"
    android:elevation="4dip"
    android:stateListAnimator="@drawable/z_selector"/>

<item android:state_pressed="true">
    <set>
        <objectAnimator android:propertyName="translationZ"
            android:duration="200"
            android:valueTo="8dip"
            android:valueType="floatType"/>
    </set>
</item>
```

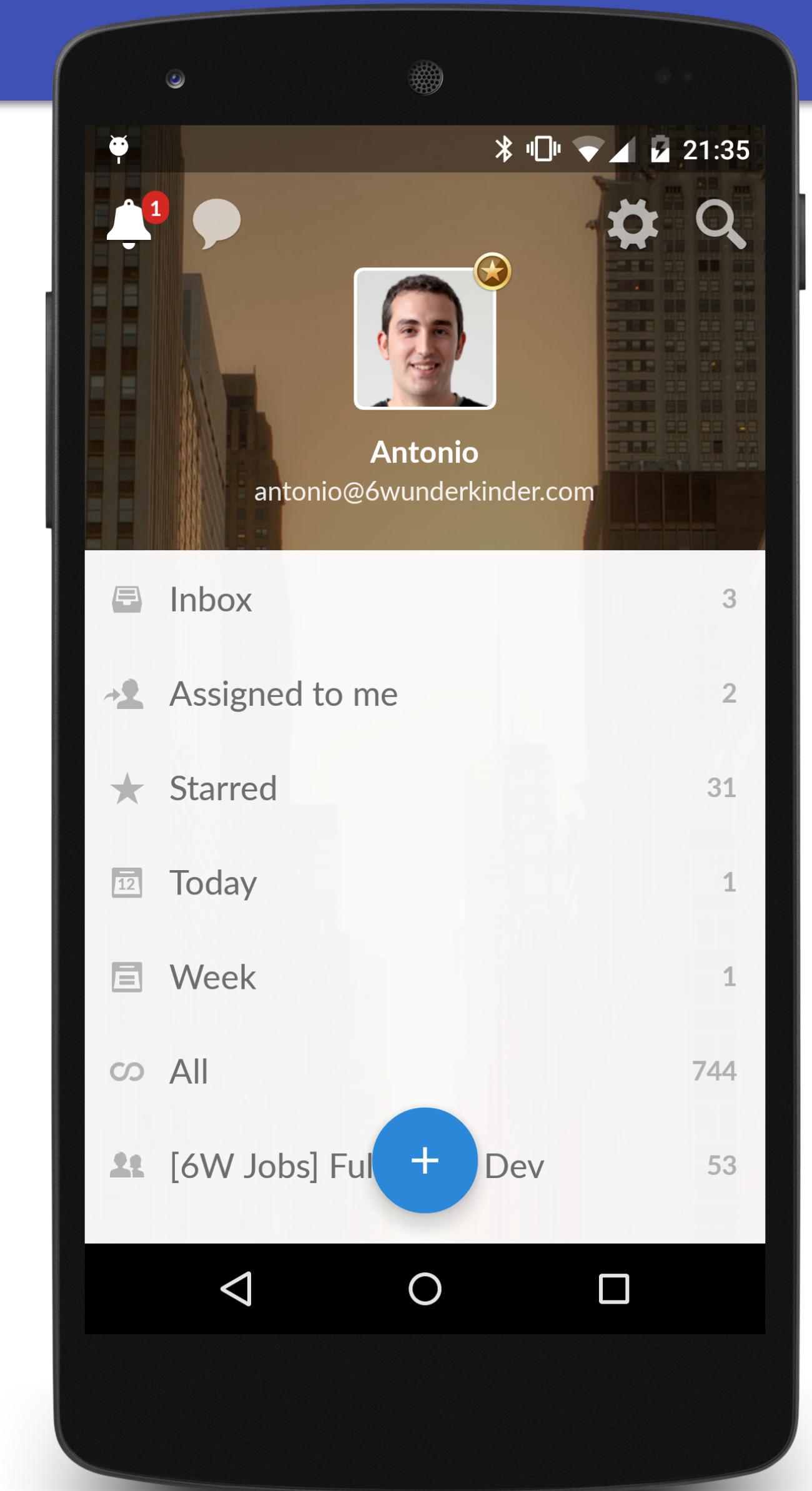


Home View - FAB

```
<ripple android:color="?attr/colorControlHighlight">
    <item android:drawable="@drawable/shape_fab_list_button"/>
</ripple>

<ImageButton
    android:layout_width="?attr/actionBarSize"
    android:layout_height="?attr/actionBarSize"
    android:layout_marginBottom="16dip"
    android:background="@drawable/fab_list_background"
    android:src="@drawable/ic_add"
    android:tint="@android:color/white"
    android:elevation="4dip"
    android:stateListAnimator="@drawable/z_selector"/>

<item android:state_pressed="true">
    <set>
        <objectAnimator android:propertyName="translationZ"
                      android:duration="200"
                      android:valueTo="8dip"
                      android:valueType="floatType"/>
    </set>
</item>
```

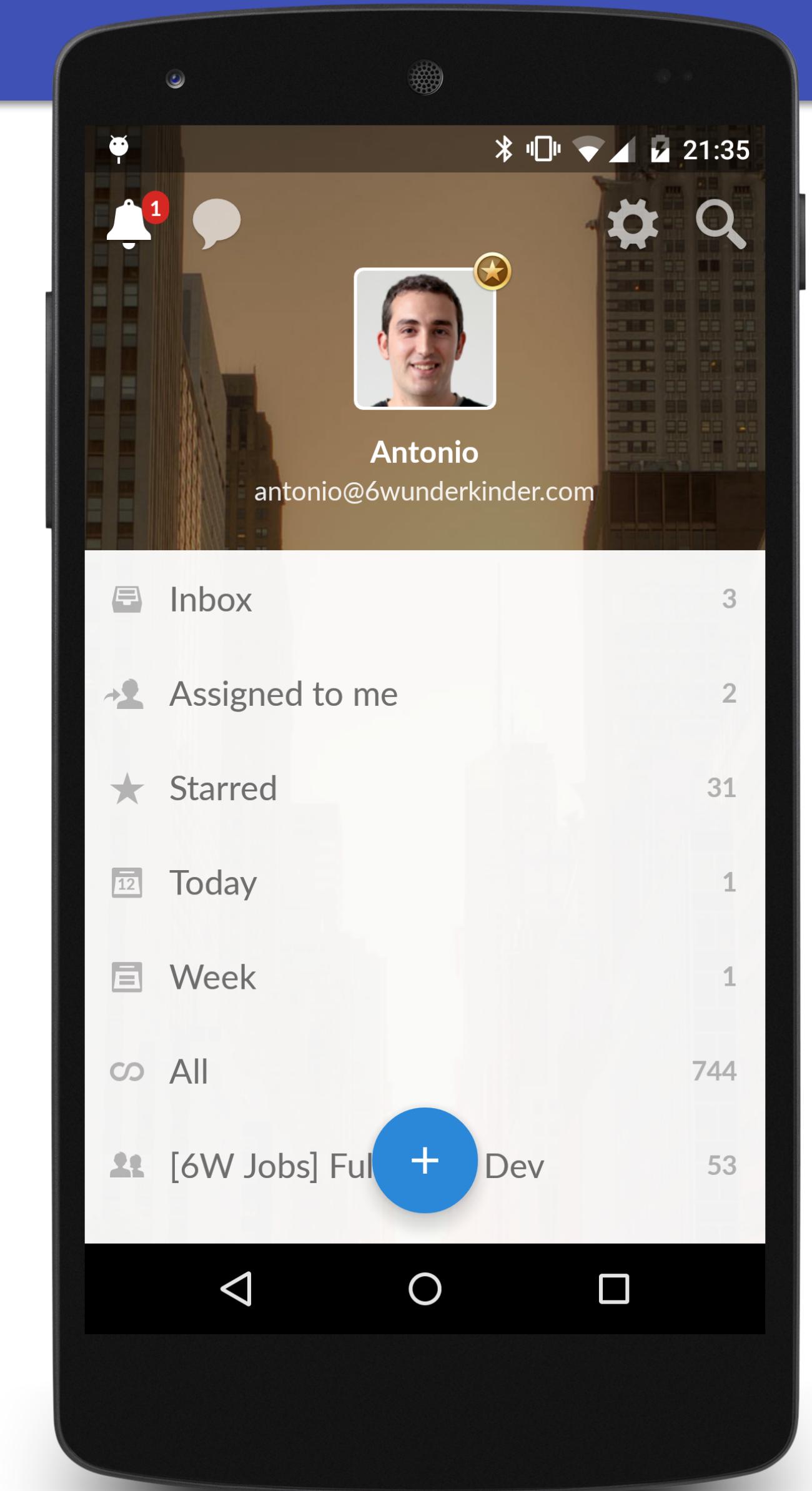


Home View - FAB

```
<ripple android:color="?attr/colorControlHighlight">
    <item android:drawable="@drawable/shape_fab_list_button"/>
</ripple>

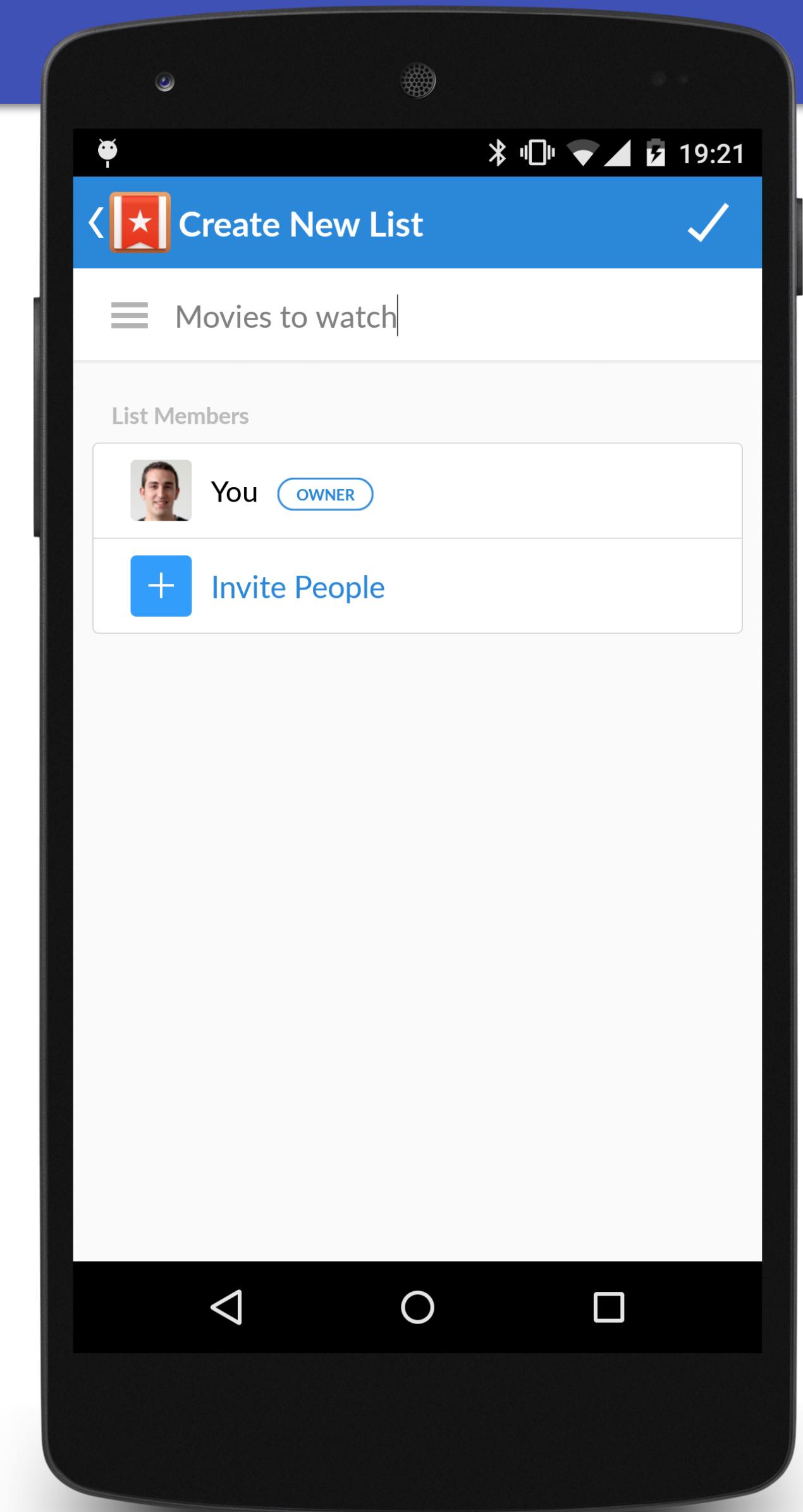
<ImageButton
    android:layout_width="?attr/actionBarSize"
    android:layout_height="?attr/actionBarSize"
    android:layout_marginBottom="16dip"
    android:background="@drawable/fab_list_background"
    android:src="@drawable/ic_add"
    android:tint="@android:color/white"
    android:elevation="4dip"
    android:stateListAnimator="@drawable/z_selector"/>

<item android:state_pressed="true">
    <set>
        <objectAnimator android:propertyName="translationZ"
            android:duration="200"
            android:valueTo="8dip"
            android:valueType="floatType"/>
    </set>
</item>
```



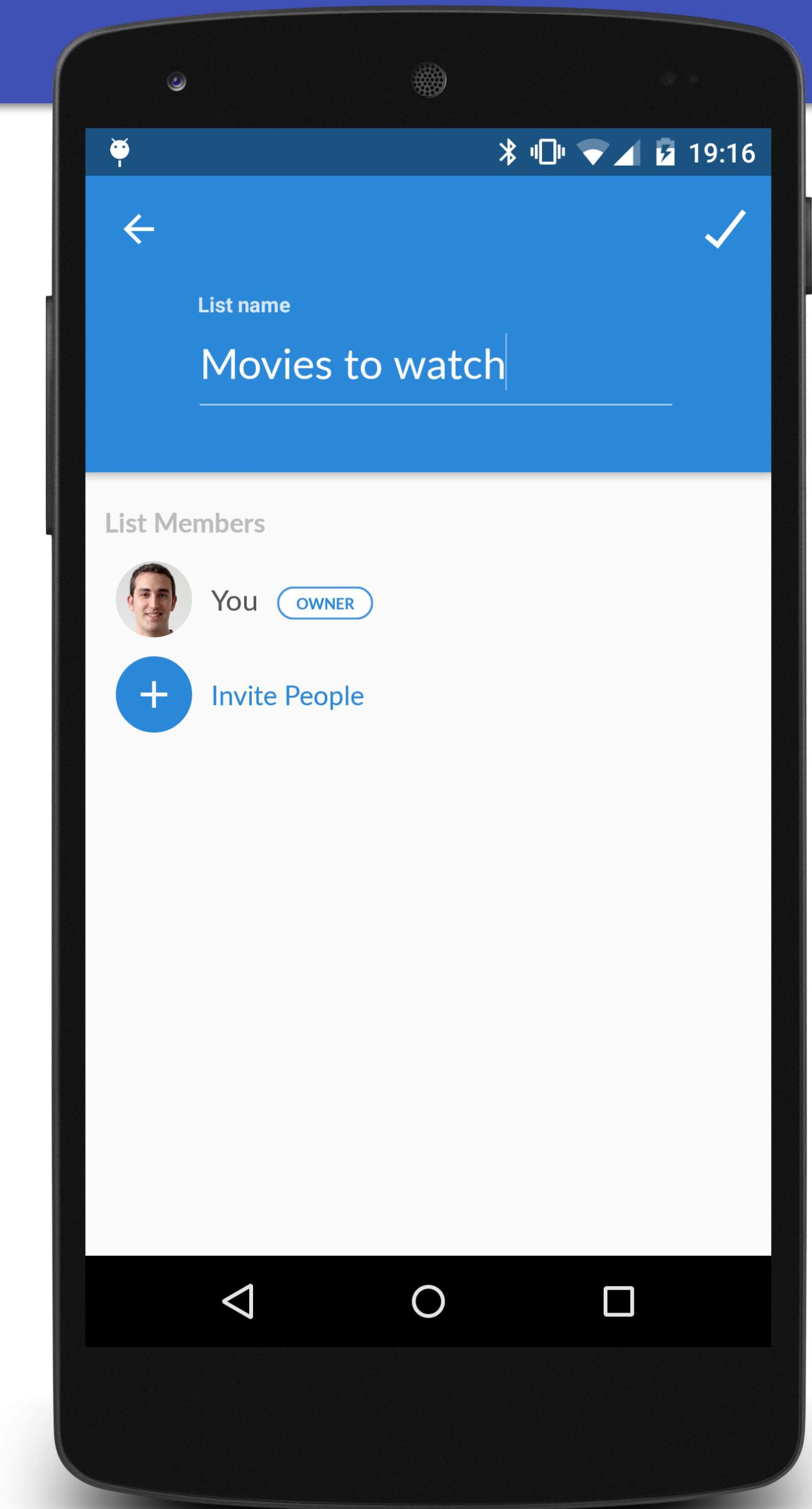
New List - Extended Toolbar

- Extended Toolbar to emphasise input field.
- Removed blocks, keep focus.
- FloatingActionButton from Chris Banes.
- And yes, rounded avatars.



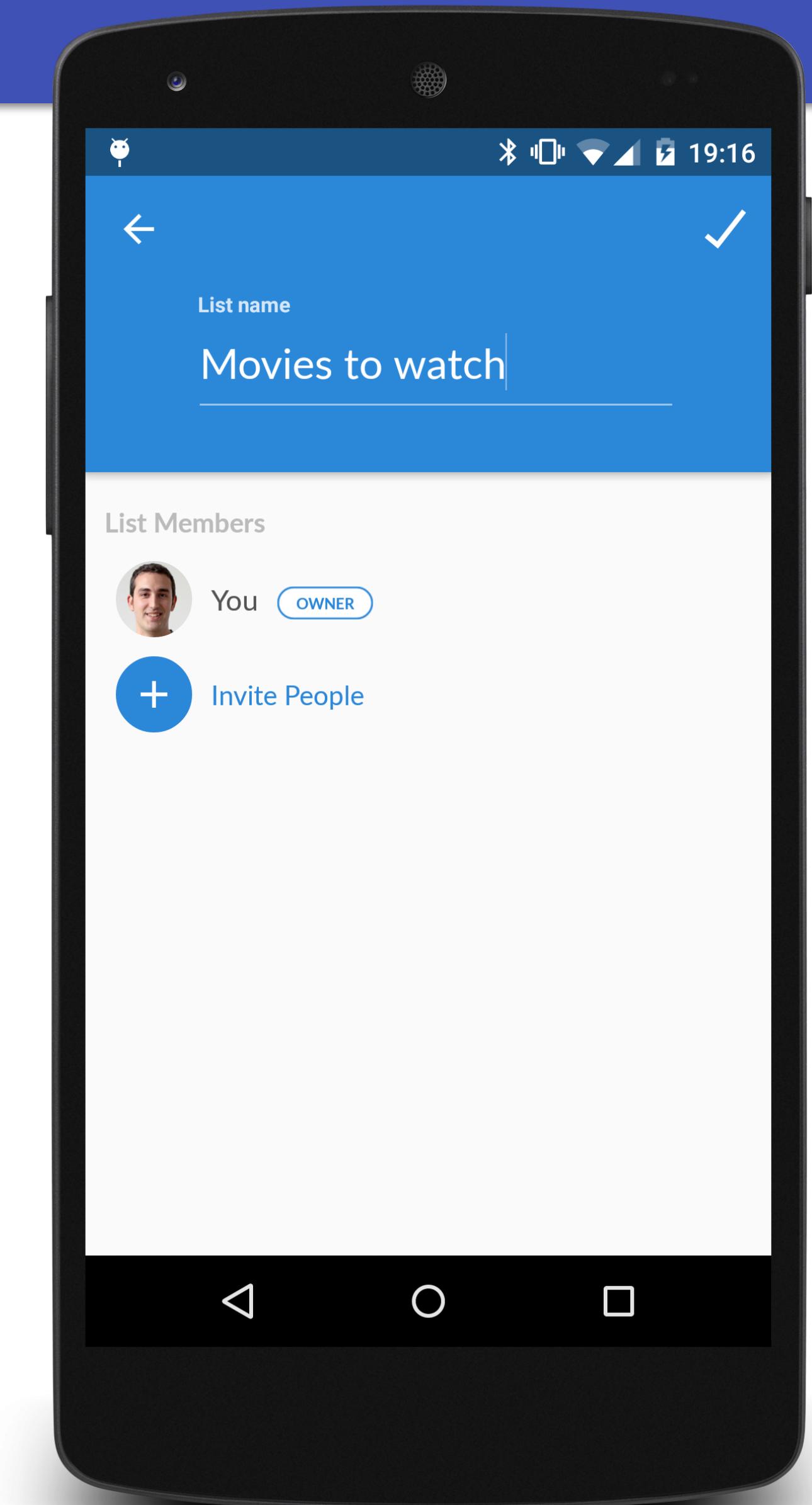
New List - Extended Toolbar

- Extended Toolbar to emphasise input field.
- Removed blocks, keep focus.
- FloatingActionButton from Chris Banes.
- And yes, rounded avatars.



New List - Extended Toolbar

- Extended Toolbar to emphasise input field.
- Removed blocks, keep focus.
- FloatingActionButton from Chris Banes.
- And yes, rounded avatars.

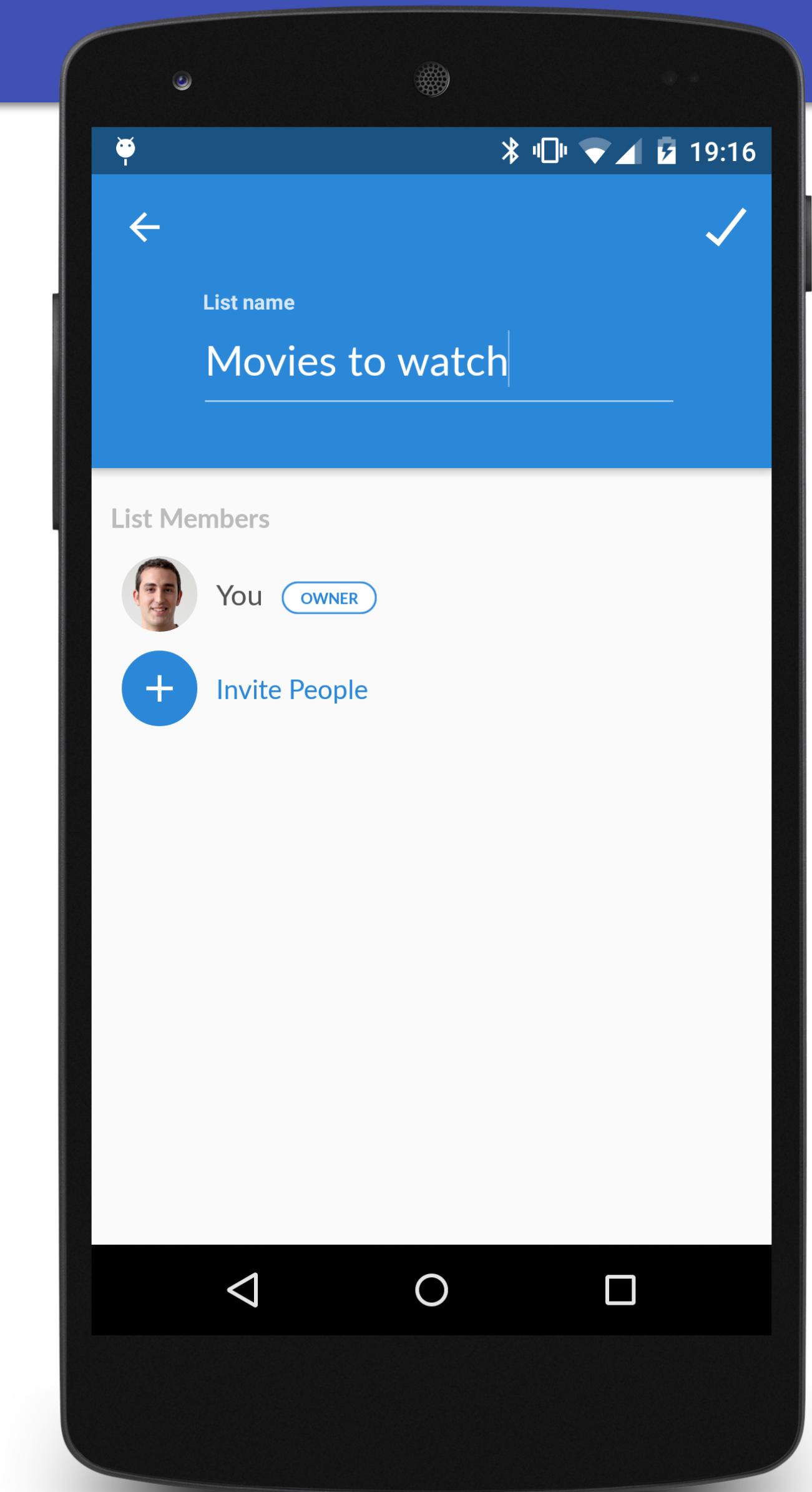


New List - Extended Toolbar

```
<android.support.v7.widget.Toolbar
    app:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
    android:elevation="4dip"
    android:minHeight="?attr/actionBarSize"
    android:layout_height="wrap_content">

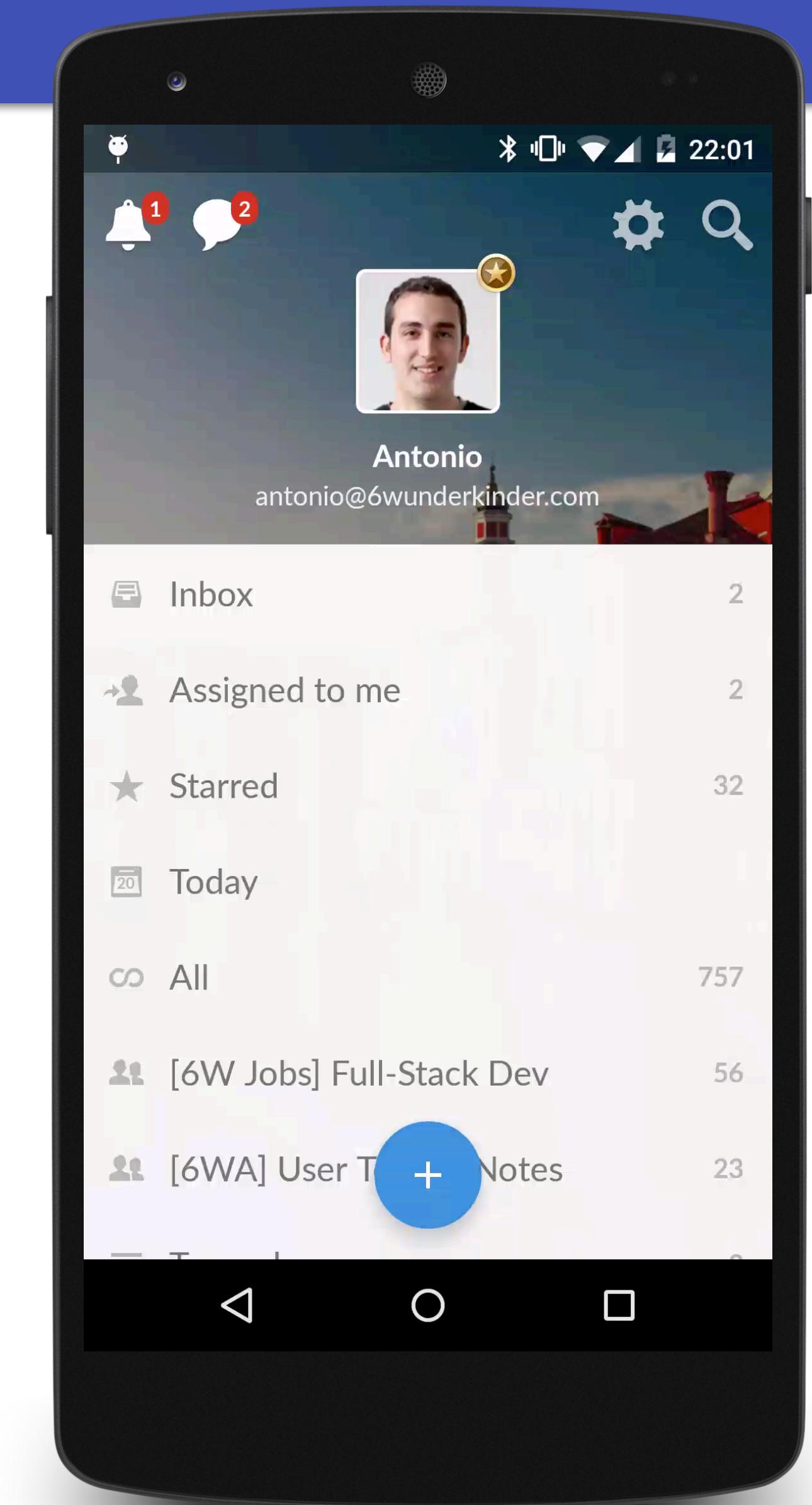
    <FloatLabelLayout
        app:floatLabelTextAppearance=
            "@style/TextAppearanceToolBar.FloatLabel"
        app:floatLabelHint="List name"
        app:floatLabelPaddingLeft="6dip">

        <EditText
            android:hint="List name"
            android:textSize="22sp"
            android:textColor="@color/white"/>
    </FloatLabelLayout>
</android.support.v7.widget.Toolbar>
```



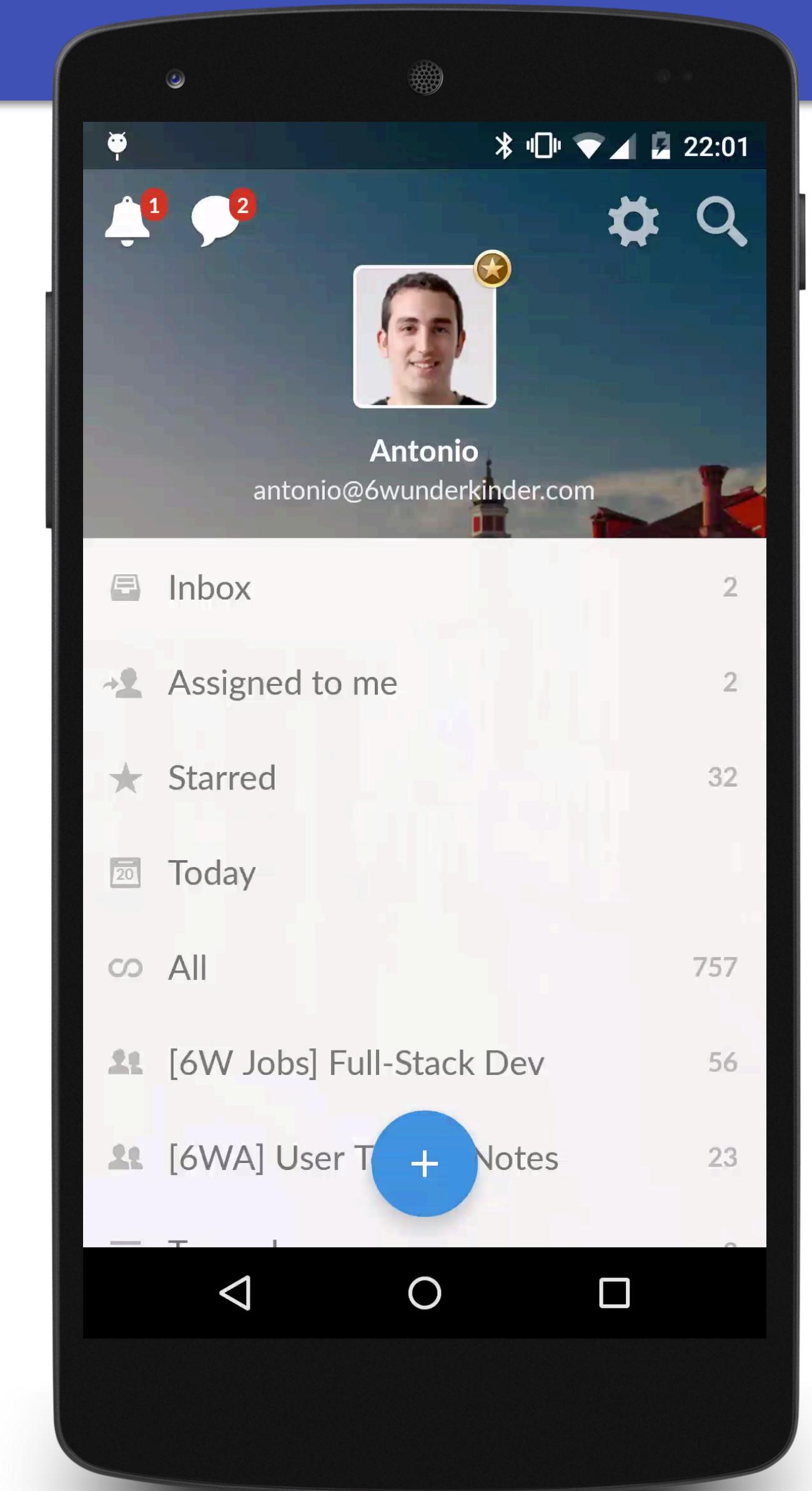
New List - Transition

- We wanted to use a meaningful transition.
- We tested 4 different variants, all with pros/cons.
- Concerns about performance in low end devices.
- FAB animation, activity transition and circular reveal.



New List - Transition

- We wanted to use a meaningful transition.
- We tested 4 different variants, all with pros/cons.
- Concerns about performance in low end devices.
- FAB animation, activity transition and circular reveal.



New List - Transition

```
ViewCompat.animate(FAB).scaleX(0.01f).scaleY(0.01f)
    .alpha(0f).withEndAction(openActivity(FAB));
```



```
<style name="WunderlistTheme.NewList">
    <item name="android:windowEnterTransition">
        @transition/new_list_transition</item>
</style>
```



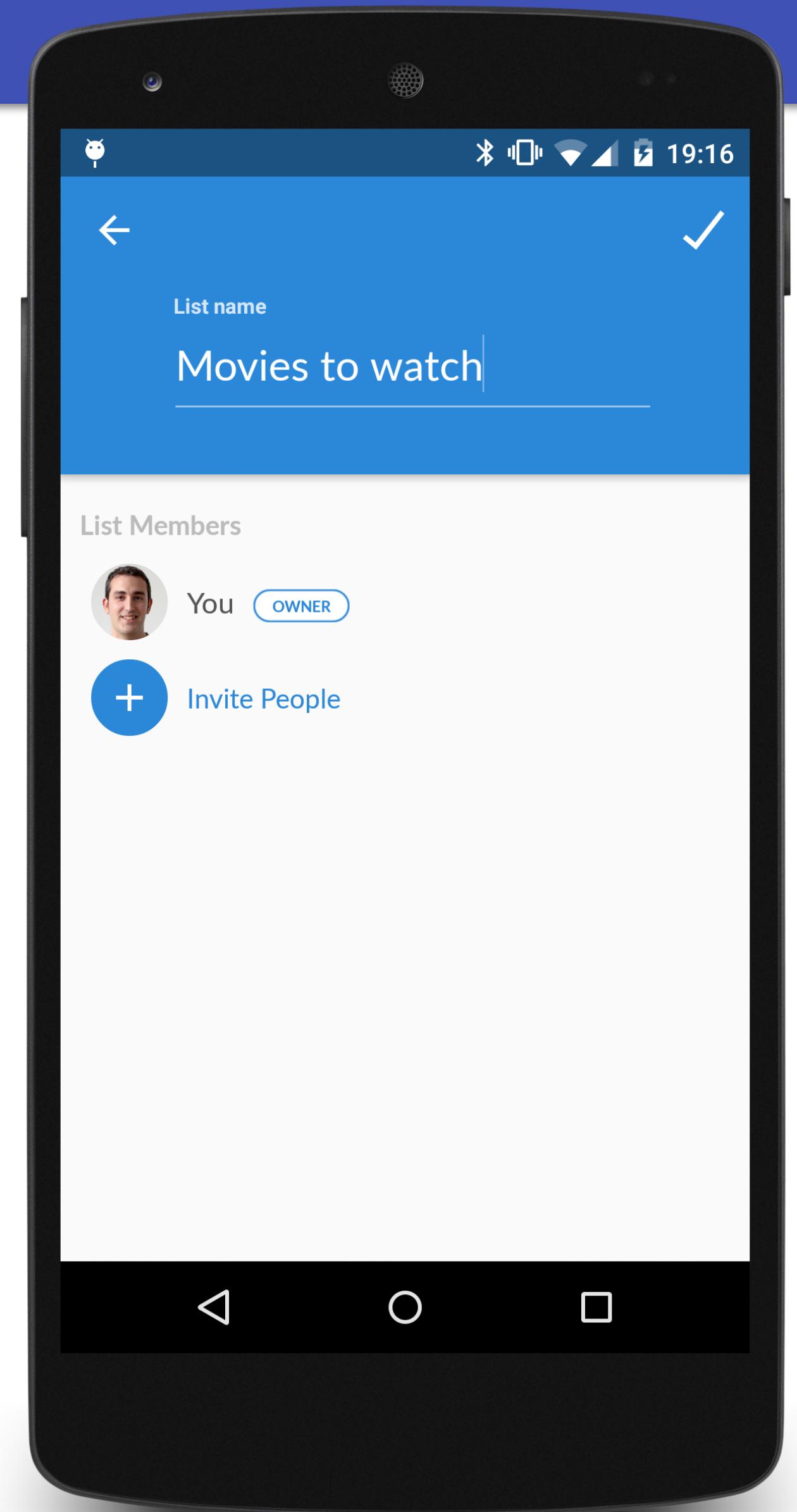
```
<transitionSet android:ordering="together">
    <explode/>
    <fade/>
</transitionSet>
```



```
int cx = (mToolbar.getLeft() + mToolbar.getRight()) / 2;
int cy = (mToolbar.getBottom() + mToolbar.getBottom()) / 2;
int finalRadius = Math.max(mToolbar.getWidth(),
                           mToolbar.getHeight());
```



```
Animator anim =
    ViewAnimationUtils.createCircularReveal(mToolbar, cx, cy,
                                             0, finalRadius);
anim.start();
```



New List - Transition

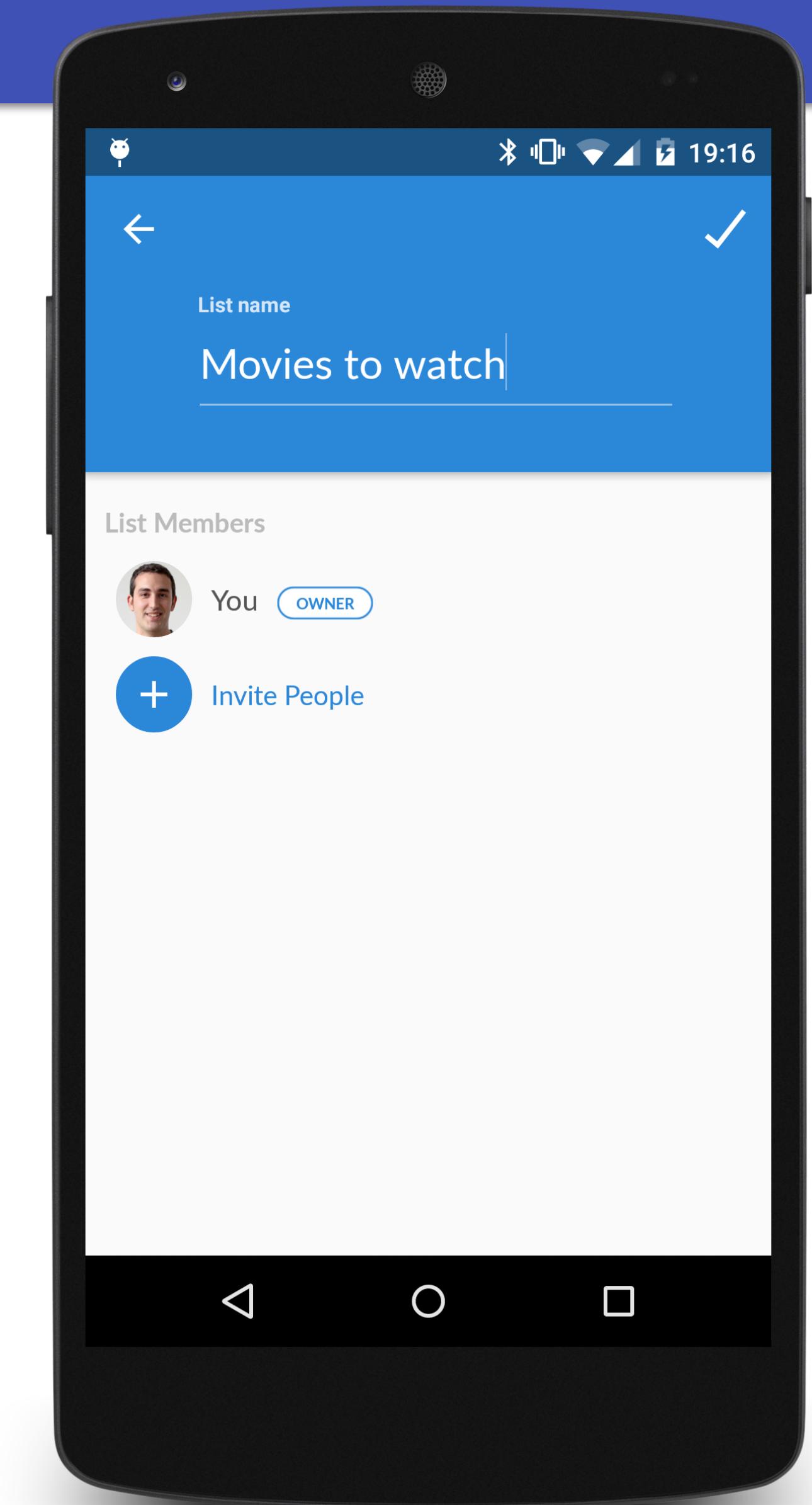
```
ViewCompat.animate(FAB).scaleX(0.01f).scaleY(0.01f)
    .alpha(0f).withEndAction(openActivity(FAB));

<style name="WunderlistTheme.NewList">
    <item name="android:windowEnterTransition">
        @transition/new_list_transition</item>
</style>

<transitionSet android:ordering="together">
    <explode/>
    <fade/>
</transitionSet>

int cx = (mToolbar.getLeft() + mToolbar.getRight()) / 2;
int cy = (mToolbar.getBottom() + mToolbar.getBottom()) /2;
int finalRadius = Math.max(mToolbar.getWidth(),
                           mToolbar.getHeight());

Animator anim =
    ViewAnimationUtils.createCircularReveal(mToolbar, cx, cy,
                                             0, finalRadius);
anim.start();
```



New List - Transition

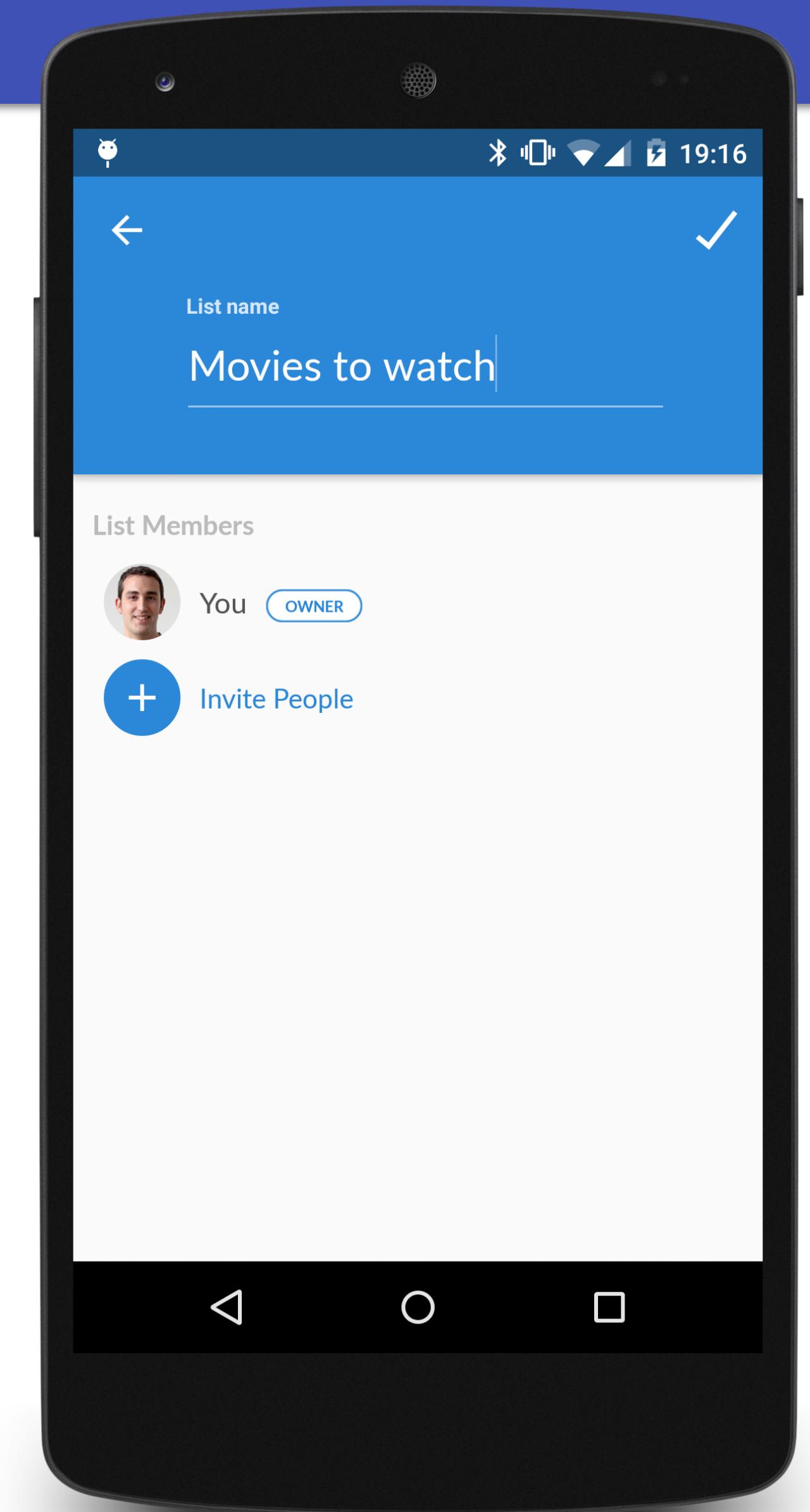
```
ViewCompat.animate(FAB).scaleX(0.01f).scaleY(0.01f)
    .alpha(0f).withEndAction(openActivity(FAB));

<style name="WunderlistTheme.NewList">
    <item name="android:windowEnterTransition">
        @transition/new_list_transition</item>
</style>

<transitionSet android:ordering="together">
    <explode/>
    <fade/>
</transitionSet>

int cx = (mToolbar.getLeft() + mToolbar.getRight()) / 2;
int cy = (mToolbar.getBottom() + mToolbar.getBottom()) /2;
int finalRadius = Math.max(mToolbar.getWidth(),
                           mToolbar.getHeight());

Animator anim =
    ViewAnimationUtils.createCircularReveal(mToolbar, cx, cy,
                                             0, finalRadius);
anim.start();
```



New List - Transition

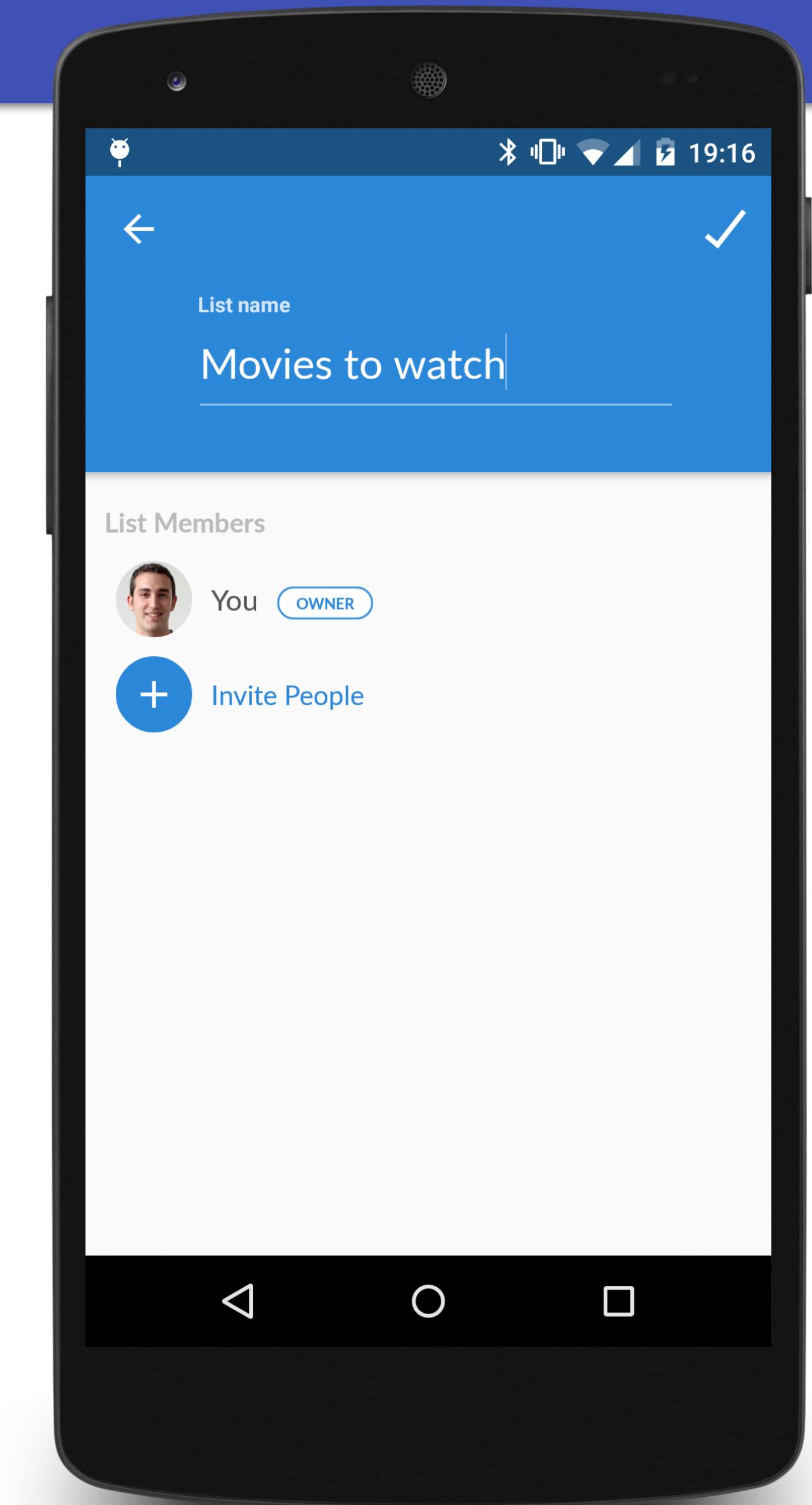
```
ViewCompat.animate(FAB).scaleX(0.01f).scaleY(0.01f)
    .alpha(0f).withEndAction(openActivity(FAB));

<style name="WunderlistTheme.NewList">
    <item name="android:windowEnterTransition">
        @transition/new_list_transition</item>
</style>

<transitionSet android:ordering="together">
    <explode/>
    <fade/>
</transitionSet>

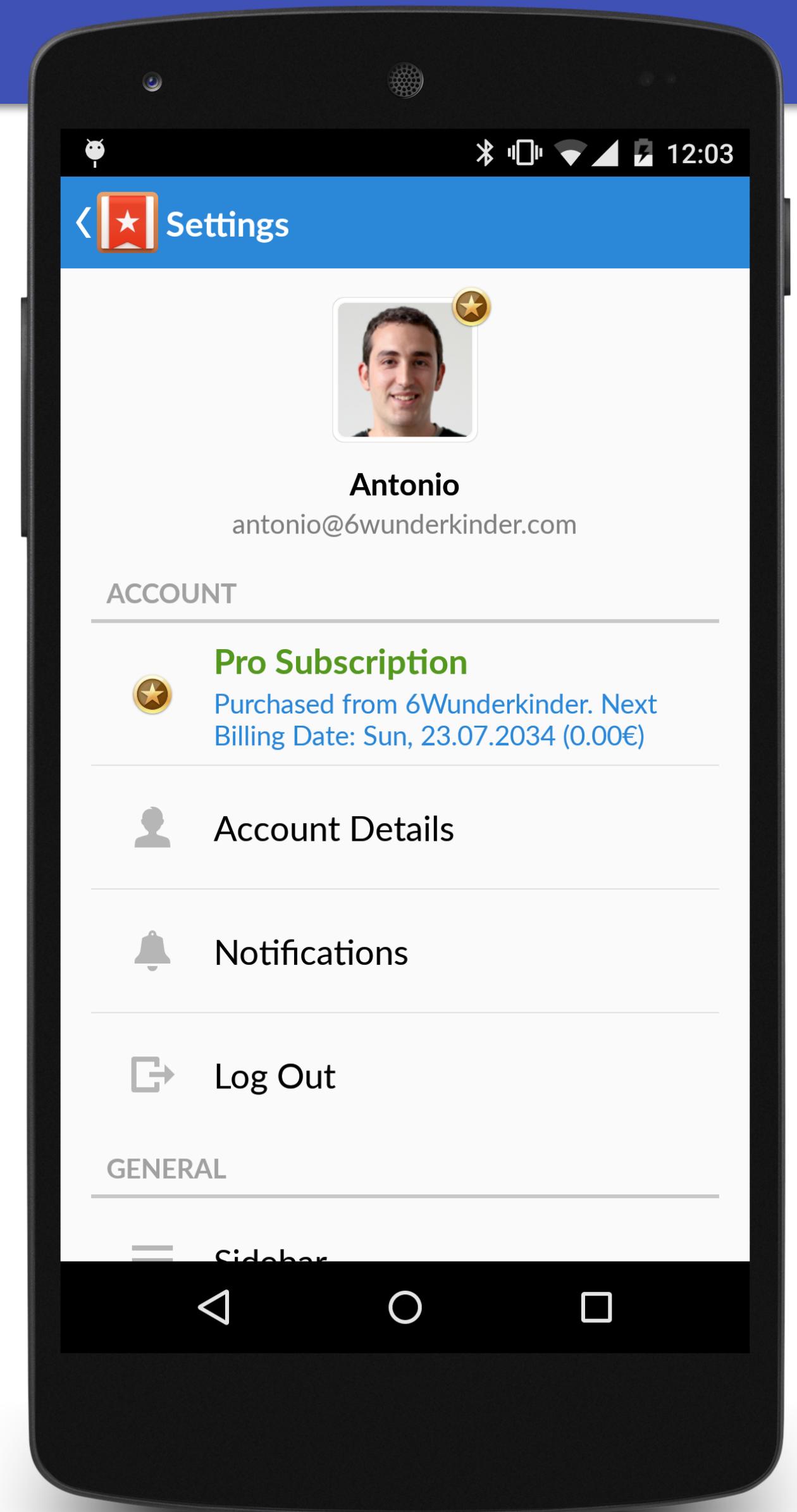
int cx = (mToolbar.getLeft() + mToolbar.getRight()) / 2;
int cy = (mToolbar.getBottom() + mToolbar.getBottom()) / 2;
int finalRadius = Math.max(mToolbar.getWidth(),
                           mToolbar.getHeight());

Animator anim =
    ViewAnimationUtils.createCircularReveal(mToolbar, cx, cy,
                                             0, finalRadius);
anim.start();
```



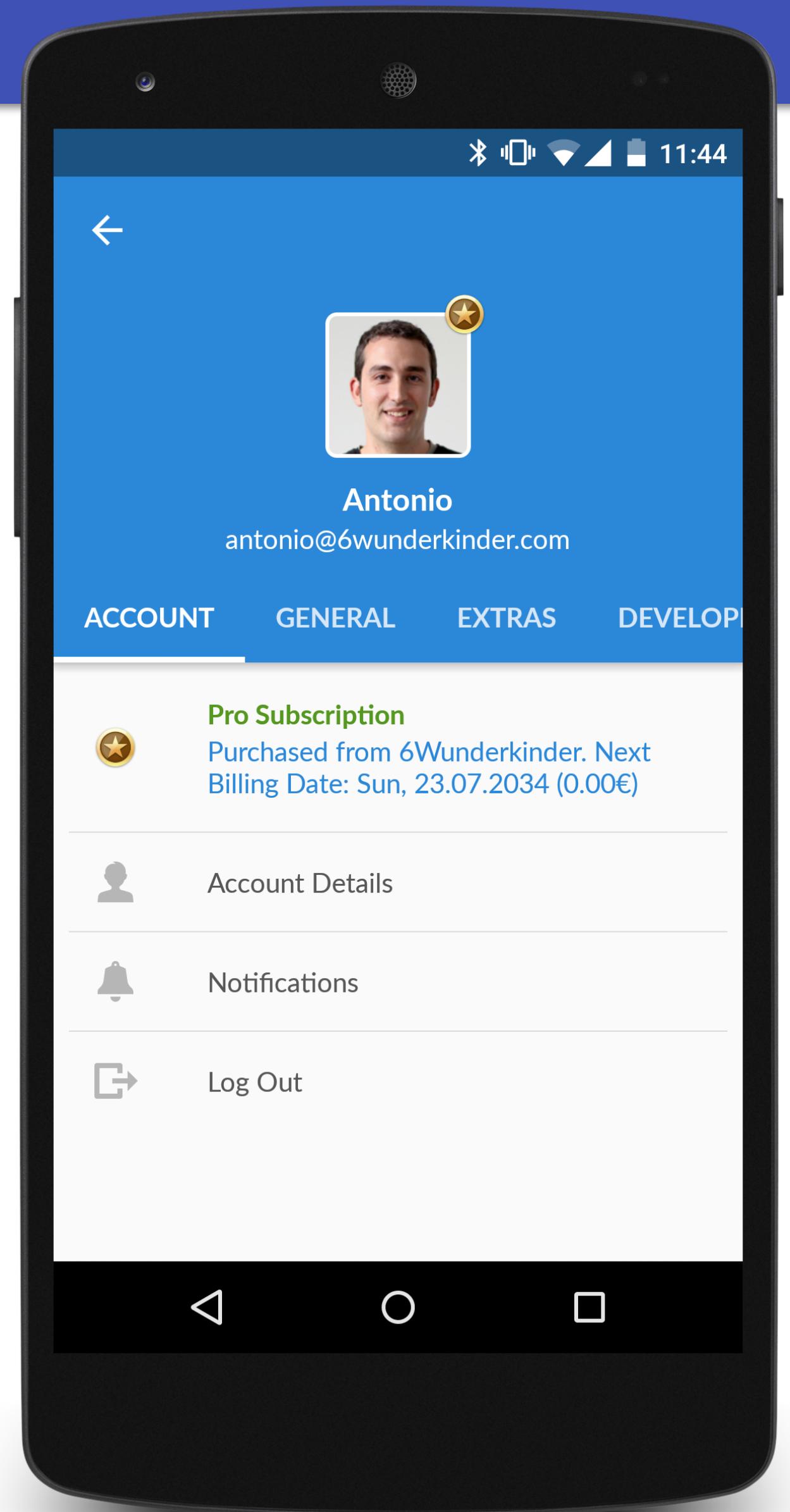
Settings

- Hard to navigate. Lots of scroll needed.
- Extended ToolBar, style from HomeView.
- ViewPager for better navigation.
- Fake headers and scroll listeners.
- Sub-sections are simply PreferenceFragment.



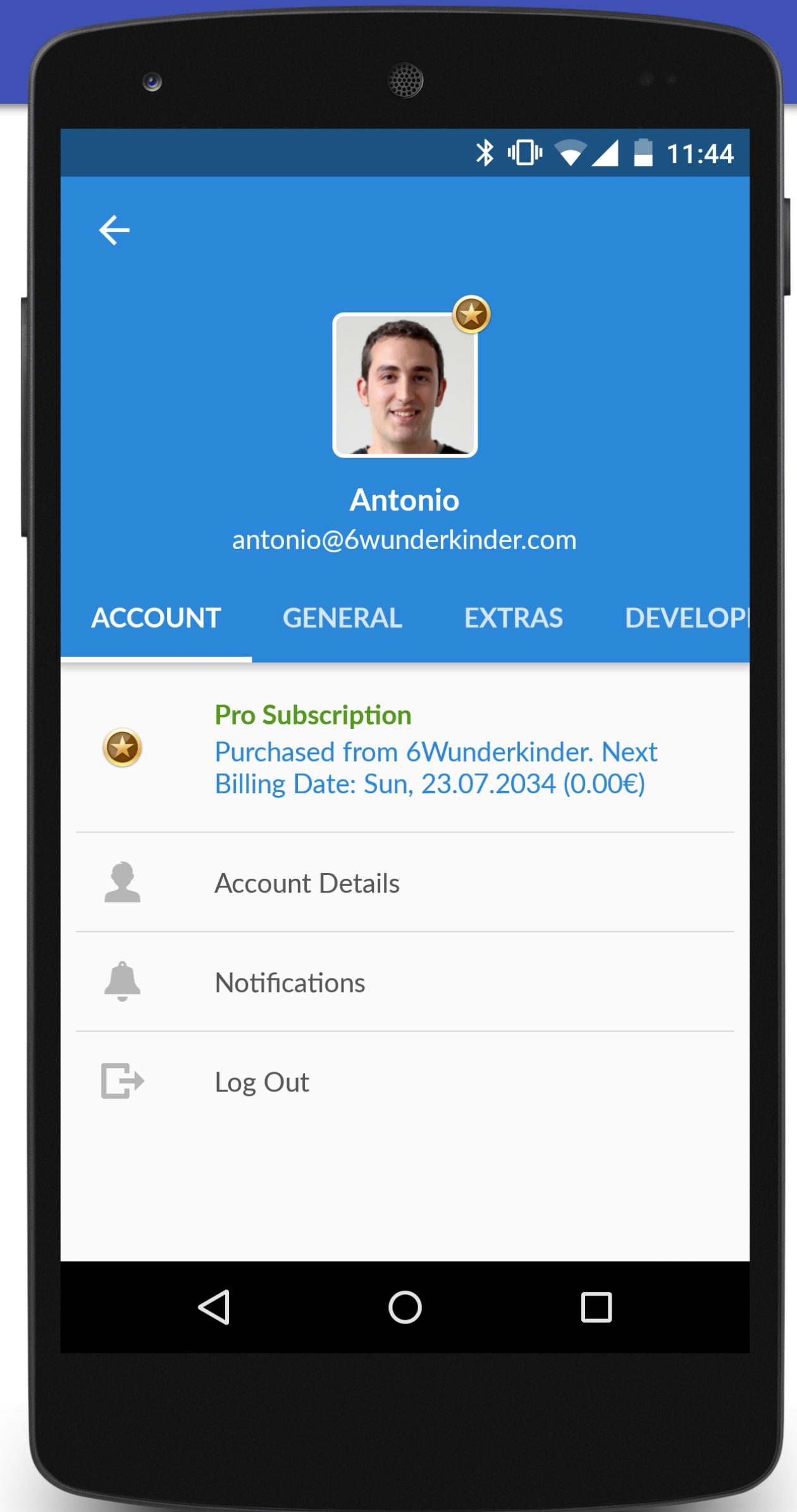
Settings

- Hard to navigate. Lots of scroll needed.
- Extended ToolBar, style from HomeView.
- ViewPager for better navigation.
- Fake headers and scroll listeners.
- Sub-sections are simply PreferenceFragment.



Settings

- Hard to navigate. Lots of scroll needed.
- Extended ToolBar, style from HomeView.
- ViewPager for better navigation.
- Fake headers and scroll listeners.
- Sub-sections are simply PreferenceFragment.



What is next?

- ~~Android Wear app.~~
- Lists hierarchy.
- Location Based Reminders.
- Integrations.
- And more!

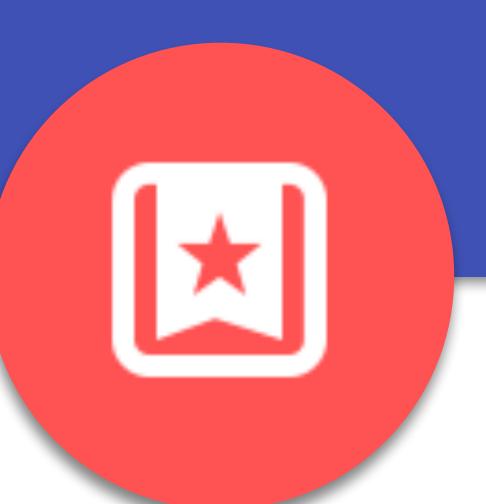




Thanks!!

Q & A

Material Fest @ Cádiz - 01/02/15



Antonio J. Consuegra
@aconsuegra

6Wunderkinder GmbH