

# Corso di Laboratorio di Programmazione per Sistemi Mobile e Tablet

Docente: Dr. Mauro Dragoni

Dipartimento di Ingegneria e Scienze dell'Informazione  
Anno Accademico 2021/2022

# in questa lezione...

- Navigazione
  - Navigazione gerarchica
    - Navigazione dei discendenti
    - Navigazione degli antenati
    - Navigazione laterale

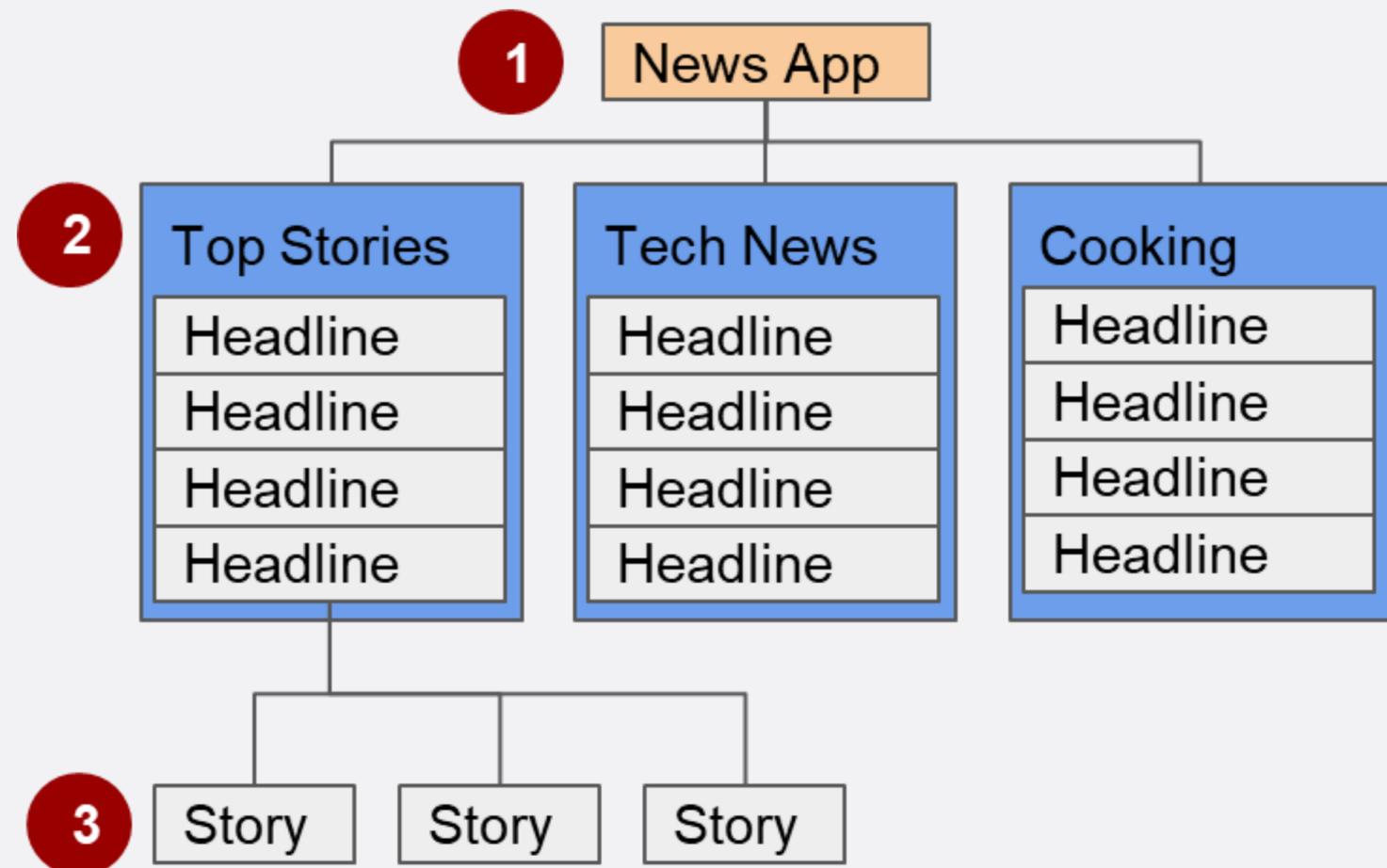
# Navigazione gerarchica

- Tre pattern principali
- **Parent screen:** navigazione da una schermata principale verso sotto-schermate.
- **Collection sibling:** navigazione verso una collezione di sotto-schermate (es. liste di elementi)
- **Section sibling:** navigazione verso i contenuti di un elemento, es. titolo->contenuto.

# Navigazione gerarchica

1. Parent
2. Collection sibling
3. Section sibling

- Best practices:
  - usare un'Activity per il parent screen
  - usare Activity o, ancora meglio, Fragment per i child screen

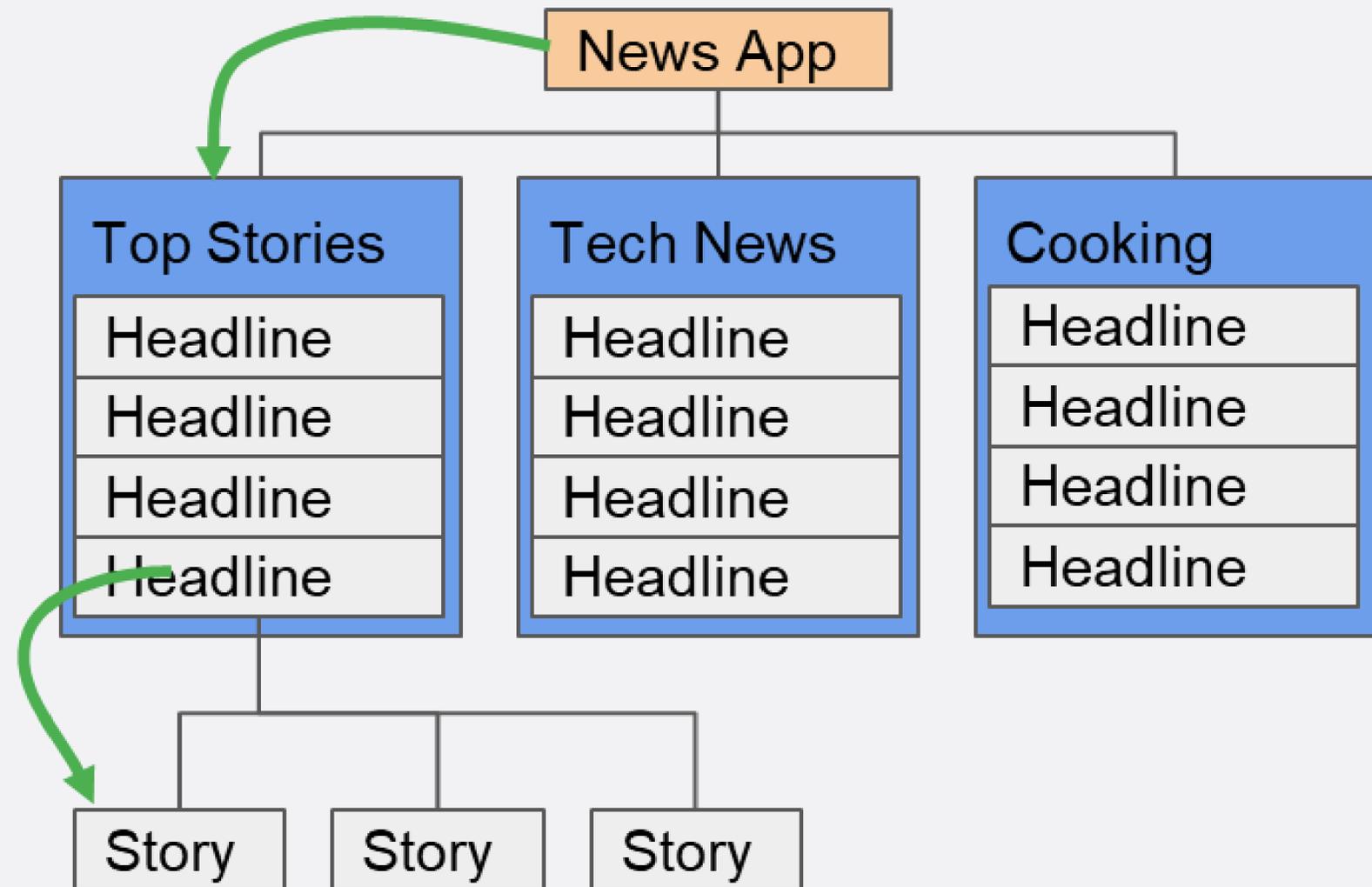


# Navigazione gerarchica

- Tipi di navigazione gerarchica
- Navigazione dei discendenti
  - da un parent ad uno dei suoi children
  - da una lista di titoli, al loro sommario, al loro contenuto effettivo
- Navigazione degli antenati
  - da uno dei children/sibling al loro parent
  - dal contenuto di un documento fino al titolo
- Navigazione laterale
  - da un sibling all'altro
  - swipe attraverso un gruppo di tabs

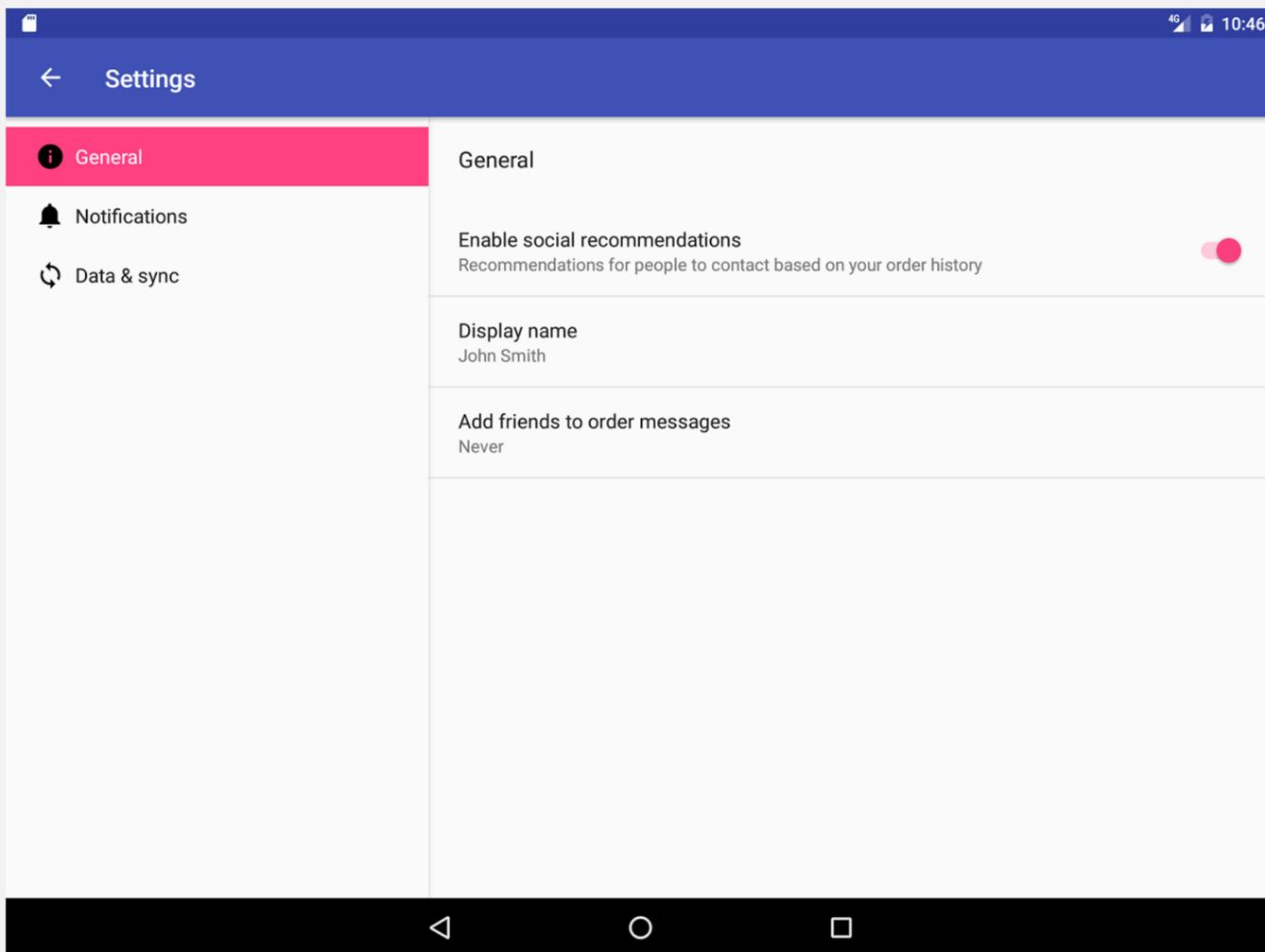
# Navigazione dei discendenti

- Da un parent ad uno dei suoi children
- Da una lista di titoli, al loro sommario, al loro contenuto effettivo

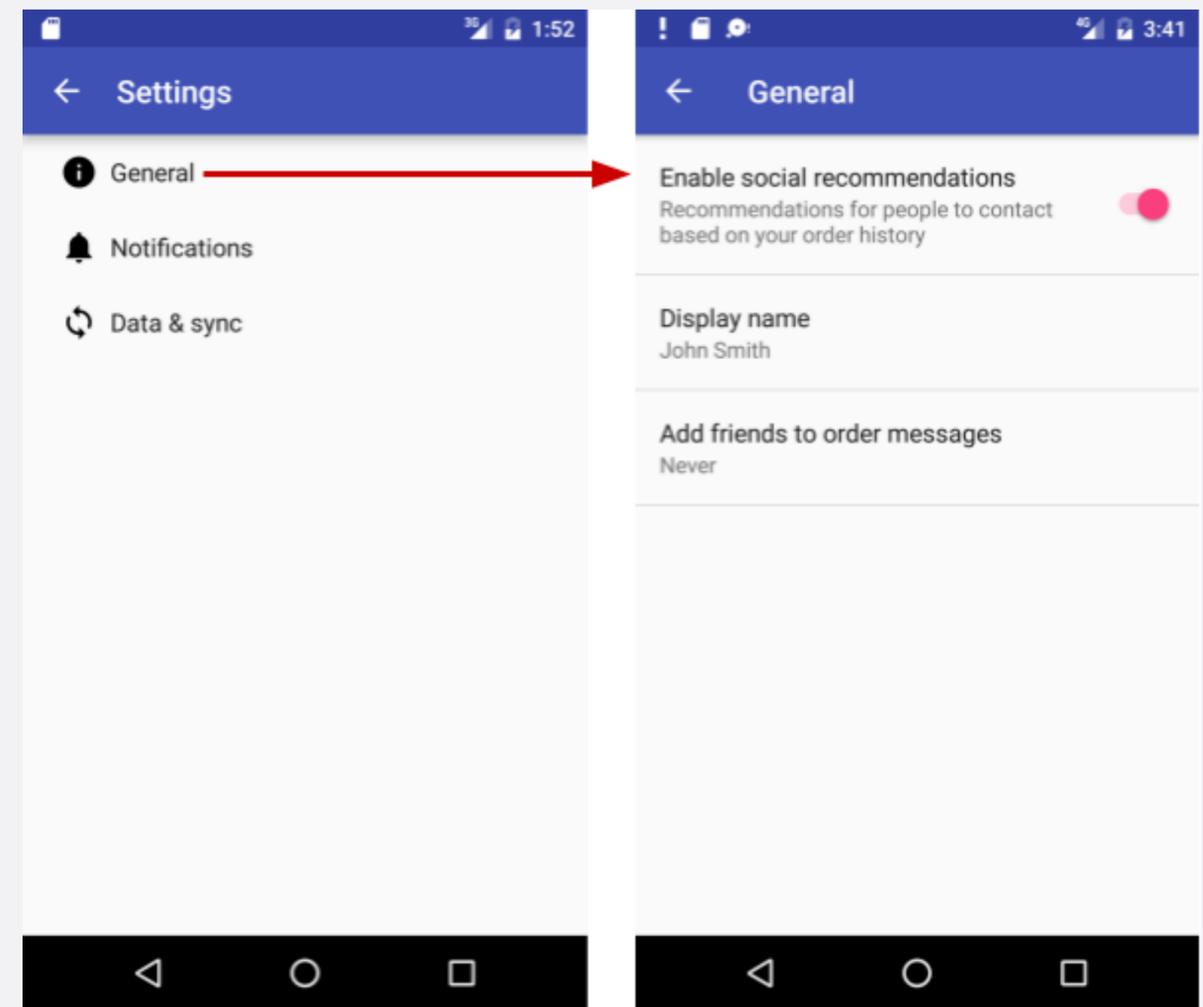


# Navigazione dei discendenti

- Side-by-side su tablet



- Screen multipli sullo smartphone

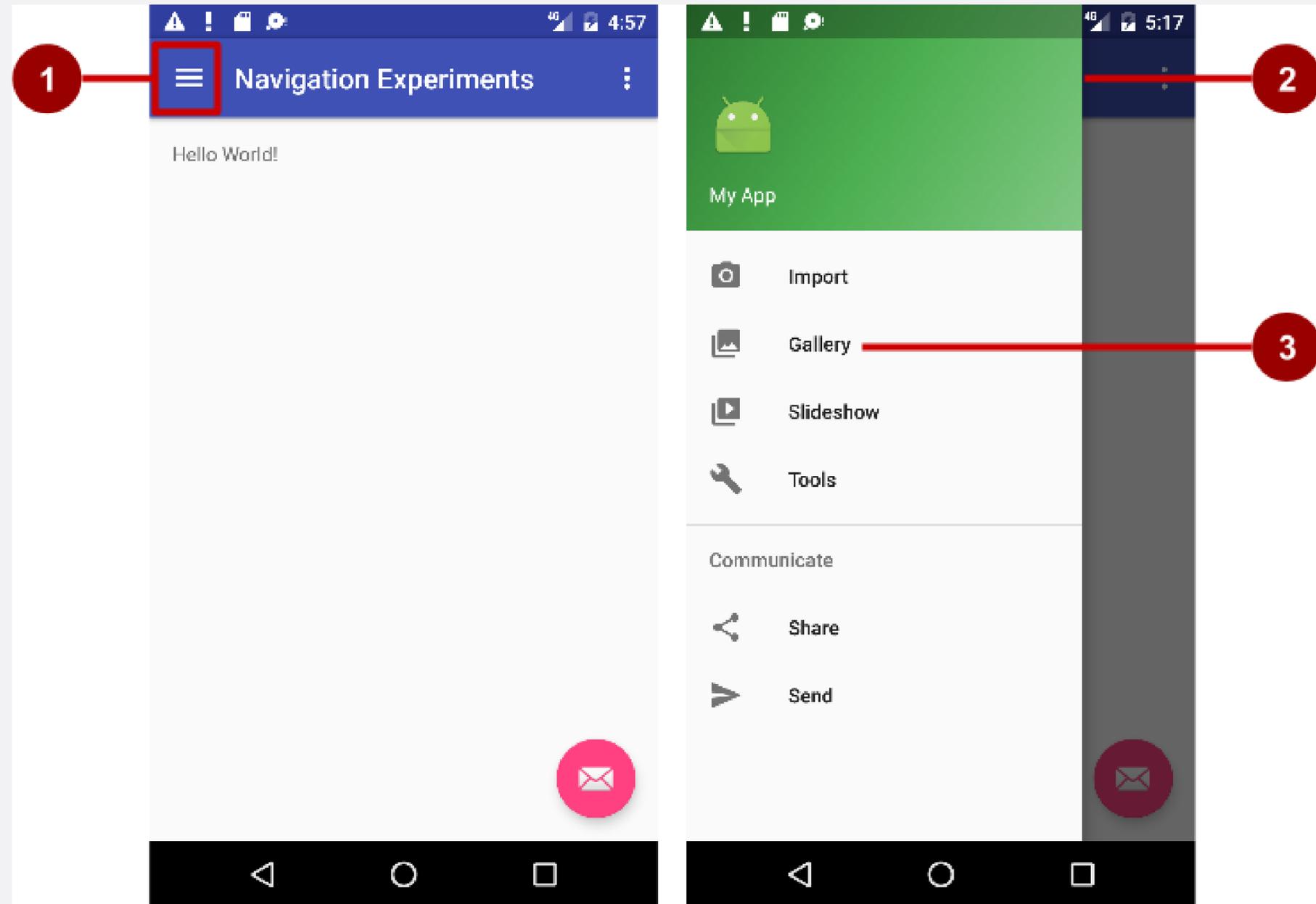


# Navigazione dei discendenti

- Controlli a supporto della navigazione dei discendenti:
  - Navigation drawer
  - Pulsanti o immagini sul layout principale
  - Altri elementi cliccabili
  - Elementi all'interno delle liste

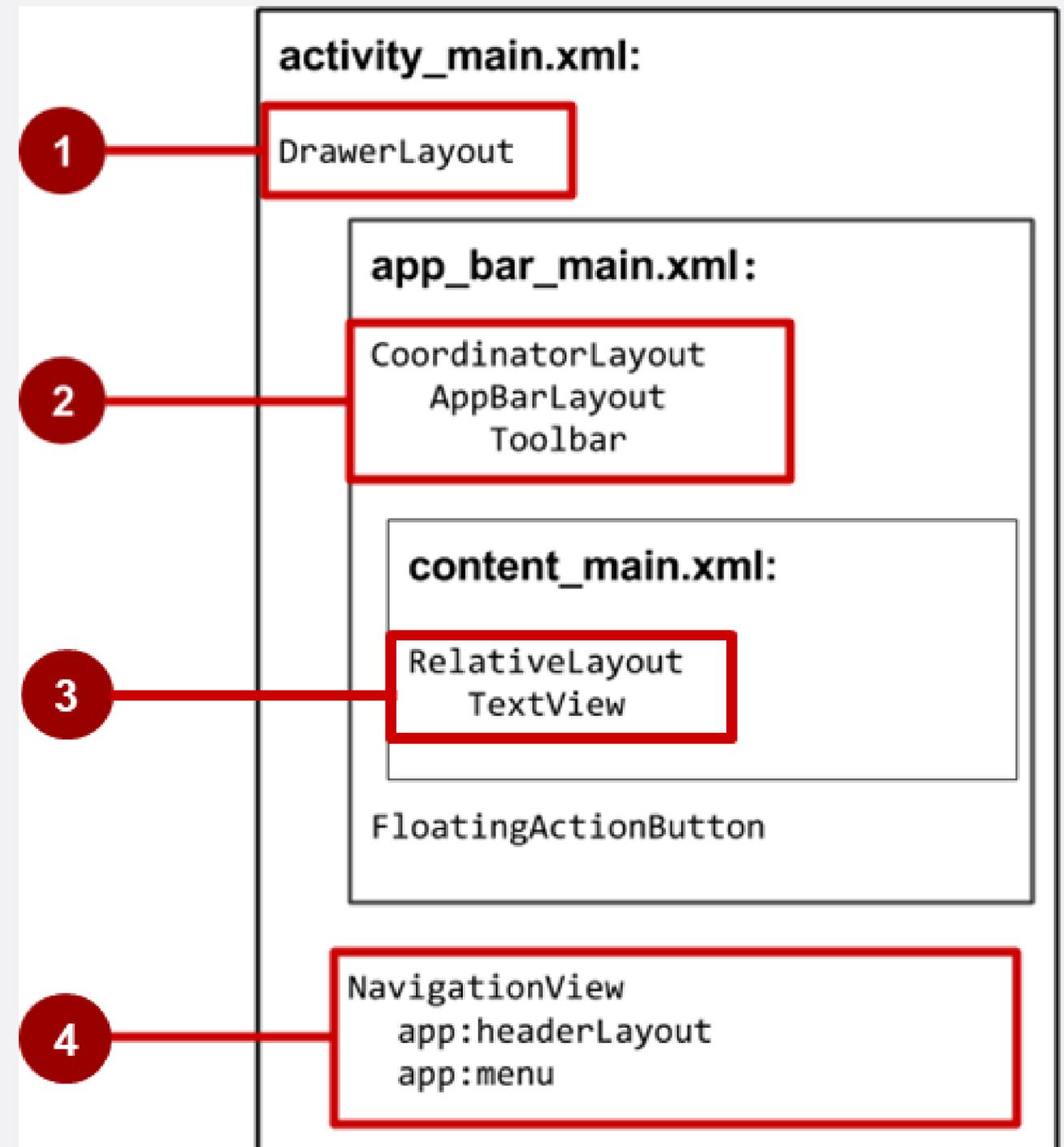
# Navigazione dei discendenti

1. Icona nella app bar
2. Header
3. Elementi del menu



# Navigazione dei discendenti

1. DrawerLayout come View principale
2. CoordinatorLayout per contenere la Toolbar
3. Layout per i contenuti della app
4. NavigationView che include i layout per lo header e gli elementi selezionabili



# Navigazione degli antenati

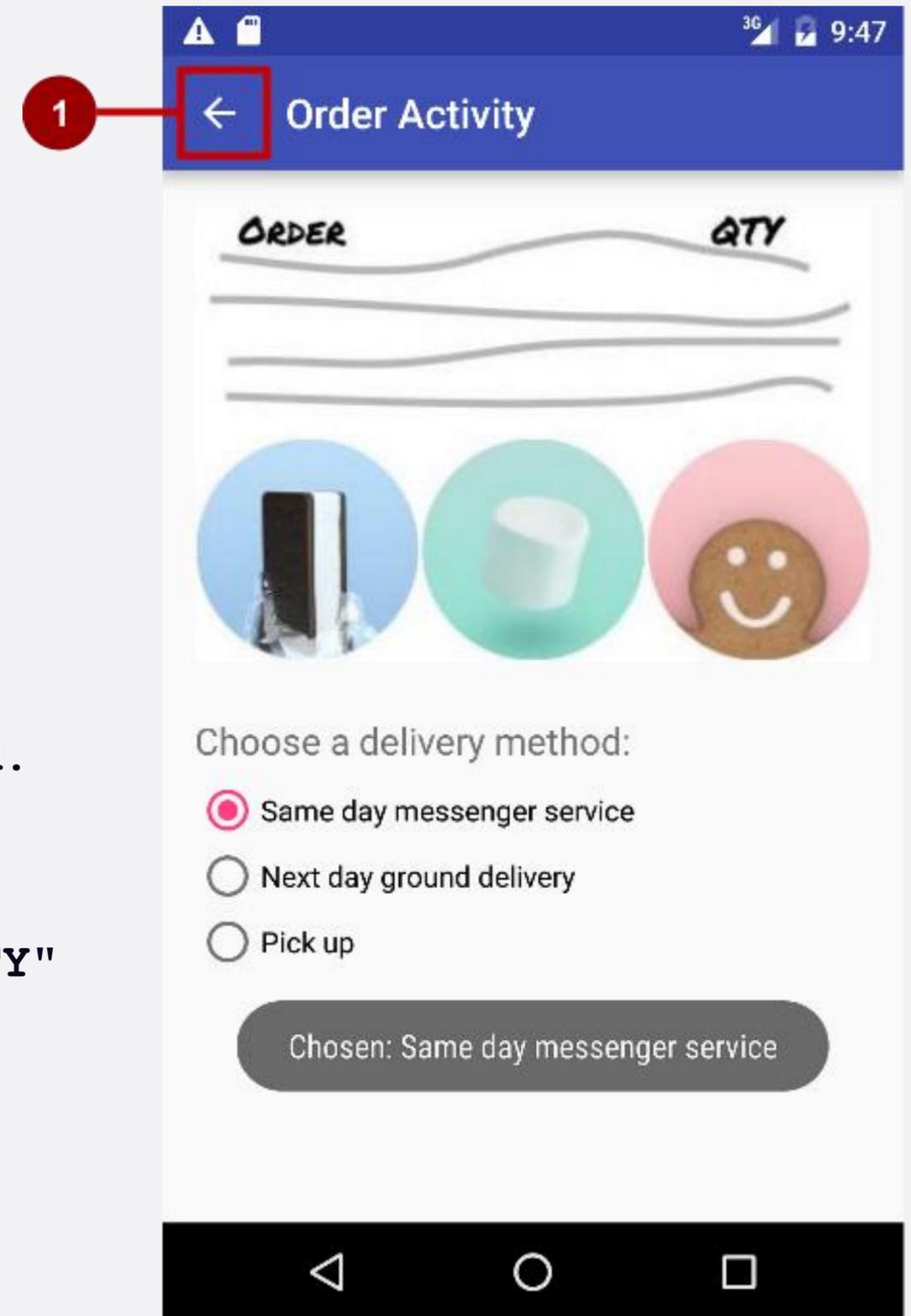
1. Permette all'utente non solo di fare un passo indietro, ma anche di salire di un livello o di aprire un parent designato.

```
<activity android:name=".OrderActivity"
  android:label="@string/title_activity_order"

  android:parentActivityName="com.example.android.
optionsmenuorderactivity.MainActivity">

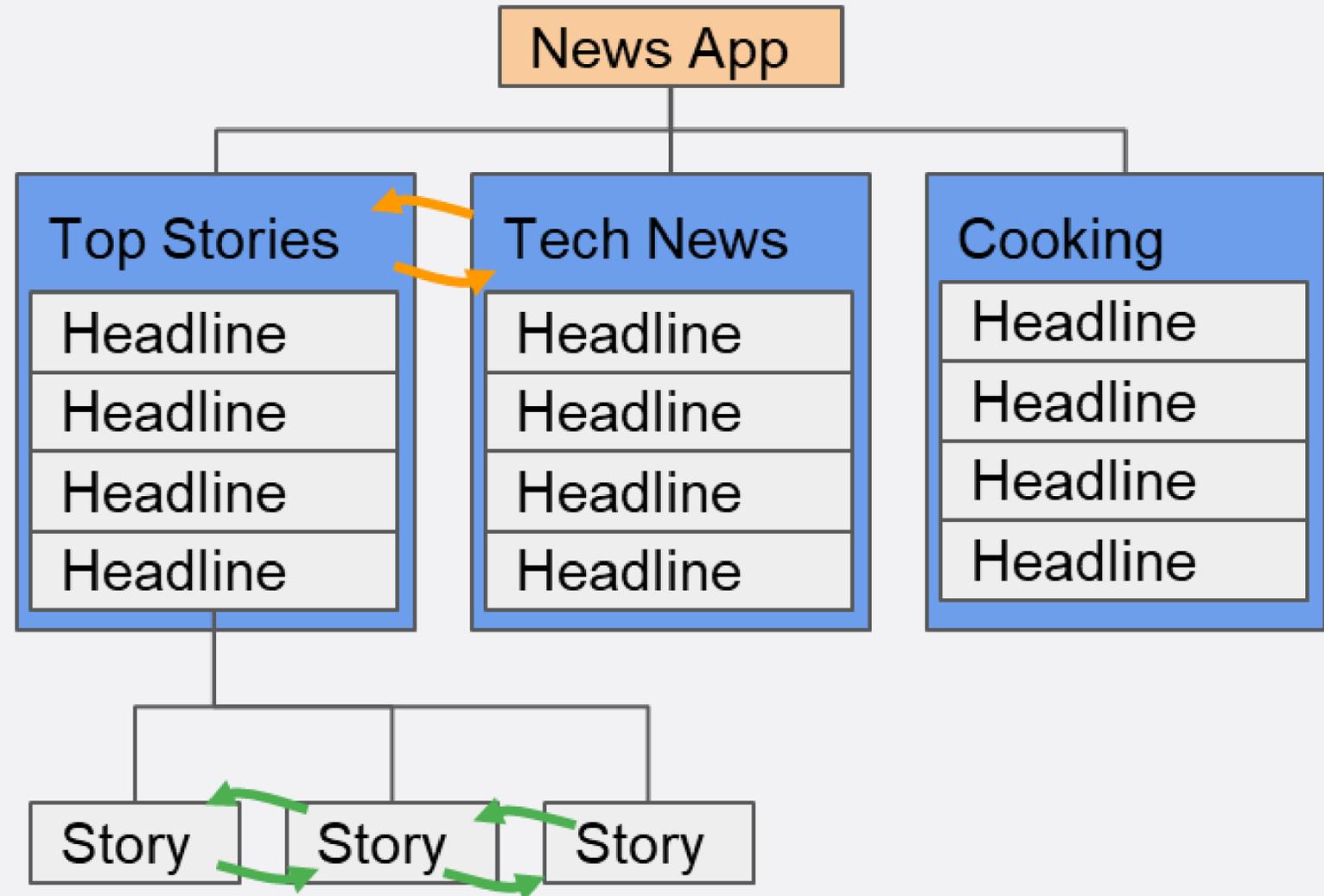
  <meta-data
    android:name="android.support.PARENT_ACTIVITY"
    android:value=".MainActivity"/>

</activity>
```



# Navigazione laterale

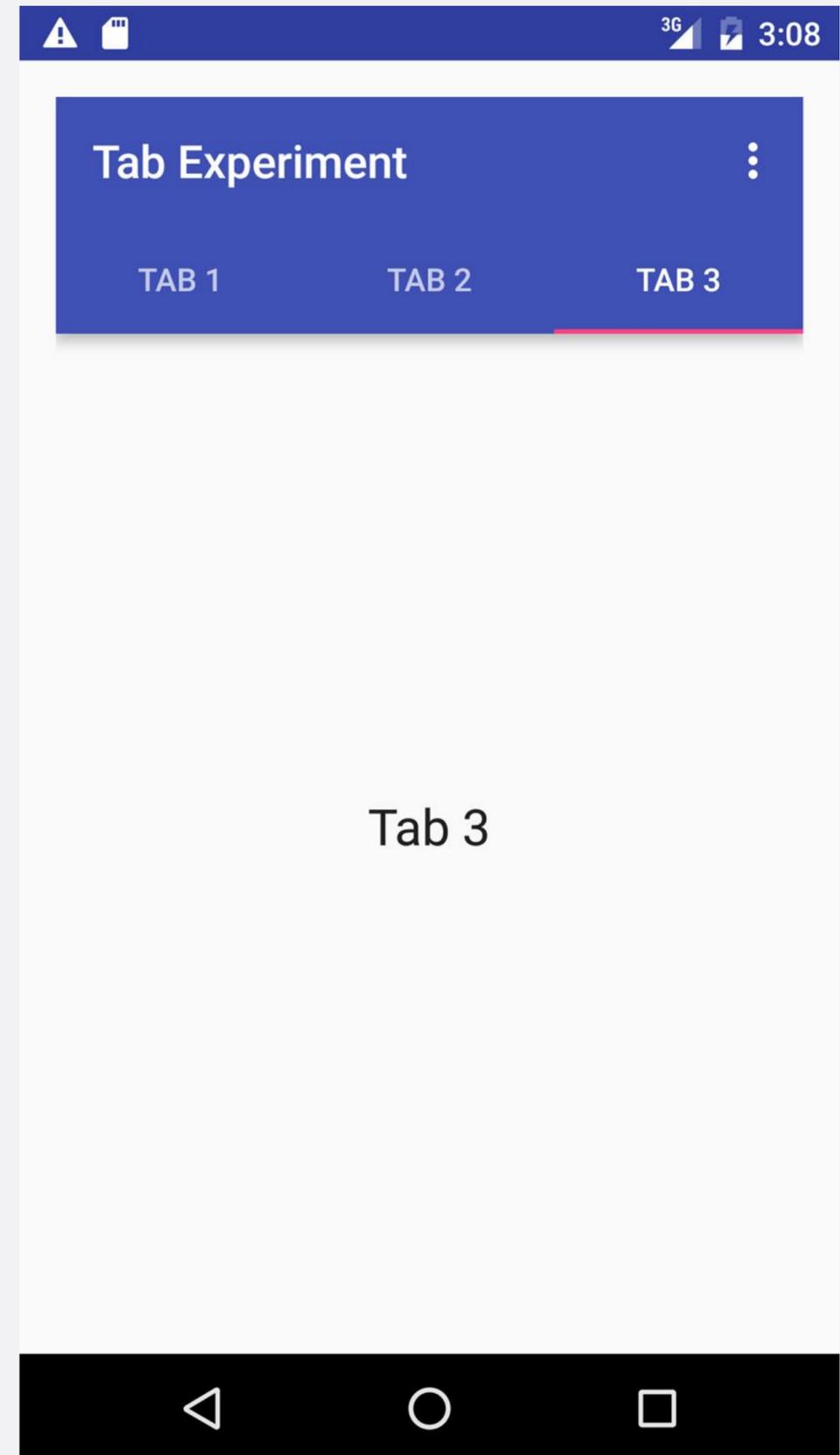
- Navigazione attraverso i sibling



# Navigazione laterale

## Benefici:

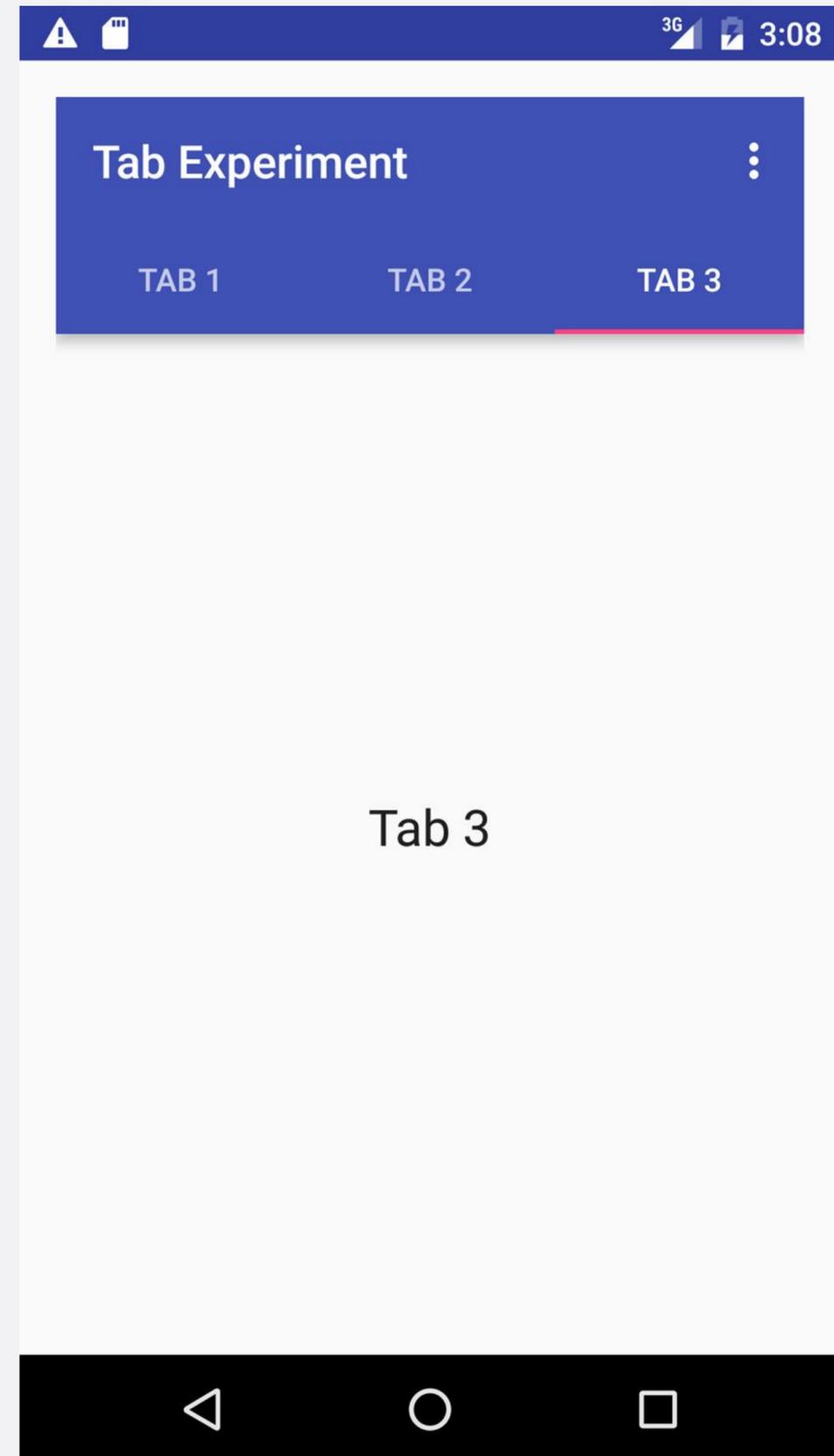
- Accesso a tutti i contenuti senza ulteriori navigazioni all'interno della app.
- Navigazione attraverso le schermate senza la necessita' di visitare il parent.



# Navigazione laterale

Best practices:

- Layout orizzontali.
- Utilizzare tutta l'area a disposizione.
- Persistenza nella struttura di visualizzazione presente in ogni tab.
- Il passaggio da un tab all'altro non deve essere utilizzato come se si volesse mostrare una storia.



# Approfondimenti

Sul sito ufficiale:

- [Navigation Design guide](#)
- [Designing effective navigation](#)
- [Creating a navigation drawer](#)
- [Creating swipe views with tabs](#)