

FPT Software

ANDROID TRAINING

LESSON 4

Version 0.1



- Fancy with Lists : customize adapter
- Option menu and Context Menu
- Inflate Xml for Menu

- Vertical scrollable view that allows multiple rows views (List item)
- Uses ListAdapter to populate each List Item
- Recycles Views (don't cache them)
- Supports different view types

- ListView Recycler:
 - ListView should not hold N Views
 - Create a custom ArrayAdapter that will inherit the Android ArrayAdapter
 - Use convertView in getView method and ViewHolder pattern



Fancy with Lists

```
@Override
public View getView(int position, View convertView, ViewGroup
    parent) {
    View row = convertView;
    SocialNetworkHolder holder = null;

    if(row == null)
    {
        LayoutInflater inflater = ((Activity)context).getLayoutInflater();
        row = inflater.inflate(layoutResourceId, parent, false);

        holder = new SocialNetworkHolder();
        holder.imgIcon = (ImageView)row.findViewById(R.id.imgIcon);
        holder.txtTitle = (TextView)row.findViewById(R.id.txtTitle);

        row.setTag(holder);
    }
    else
    {
        holder = (SocialNetworkHolder)row.getTag();
    }

    SocialNetwork weather = data[position];
    holder.txtTitle.setText(weather.title);
    holder.imgIcon.setImageResource(weather.icon);

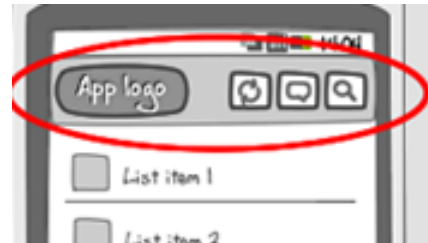
    return row;
}
```

```
static class SocialNetworkHolder
{
    ImageView imgIcon;
    TextView txtTitle;
}
```

New UI design patterns

- http://www.androidpatterns.com/uap_category/navigation

- **Action Bar**



- is located at the top of the screen to support navigation and highlight important functionalities
- replaces the title bar (which is often included into it)
- is best used for actions across your app, like search, refresh and compose
- can provide a quick link to app home by tapping the app logo
- is preferably not contextual, but actions can differ from page to page

- **Use when**

- You want to bring key actions on screen, so they are in sight. You don't want to hide them under the menu-button. The action bar is best used for common actions like search, refresh and compose. When screen real estate is valuable, consider putting actions in the options menu.



- **Category Dashboard**

- On a category dashboard, entrance points to content are displayed in several categories. The categories are represented by an icon and a title, and lie full screen in a grid orientation. This lay-out allows the user to find content faster.

- **Use when**

- A category dashboard can be used when your app is content-focused and your content can be divided into categories. To improve searchability, a search bar can be incorporated into the dashboard. The dashboard can also be used together with an action bar.

- **Dashboard Features**



- A features dashboard lays out the key functionality of your app full screen. It provides easy access to important tasks and functions. Features are displayed as an icon plus title in a grid lay-out.

- **Use when**

- A features dashboard can be used when your application is task-oriented and supports multiple tasks or features. A dashboard is ideal if you want to provide a quick overview of interesting, new, or frequently used tasks. A dashboard can be paired with an action bar, e.g. to provide a search option for users.

- **The Good**

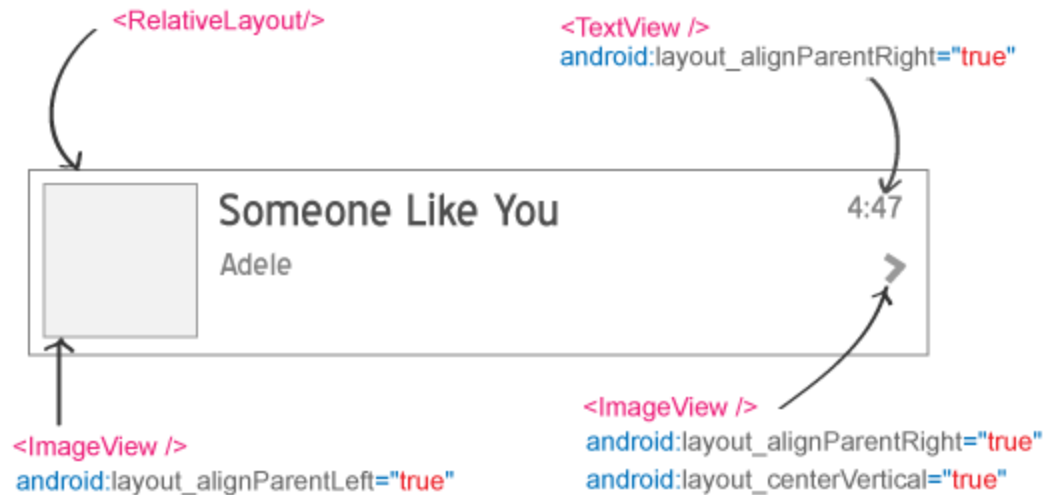
- A user has easy access to the most important tasks
- The dashboard gives an overview of the application's functionality
- If customizable can be a set of key features the user what quick access to. Good for intermediate to expert users, because they roughly know what to do when they launch the app.

Dashboard and Action Bar

- **The Bad**

- A dashboard takes up a lot of screen real estate
- The features displayed in the dashboard can be perceived as the only features of the app
- A dashboard can present an extra click to get to the actual data
- Not easy to develop categories with gridView...
- Slow navigation switching tasks. Have to go to dashboard and then choose the desired task/view. An app should start in the most frequent view

- Create a custom list view with itemview:



Thank you!