

User Preferences

Renault@lrde.epita.fr



Configuration of your Application



A dedicated screen to configure the application



Great user experience



Can be built by-hand



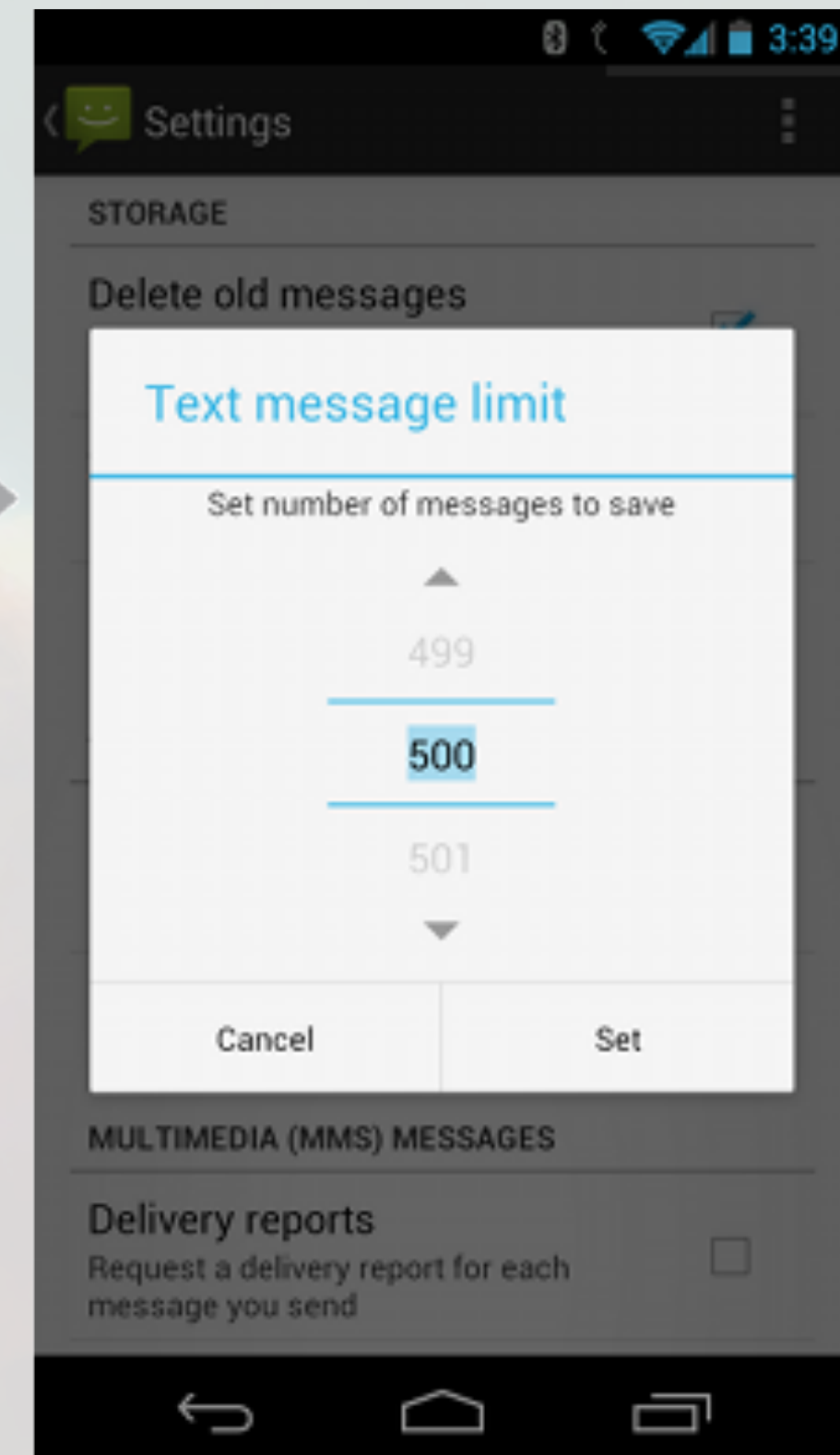
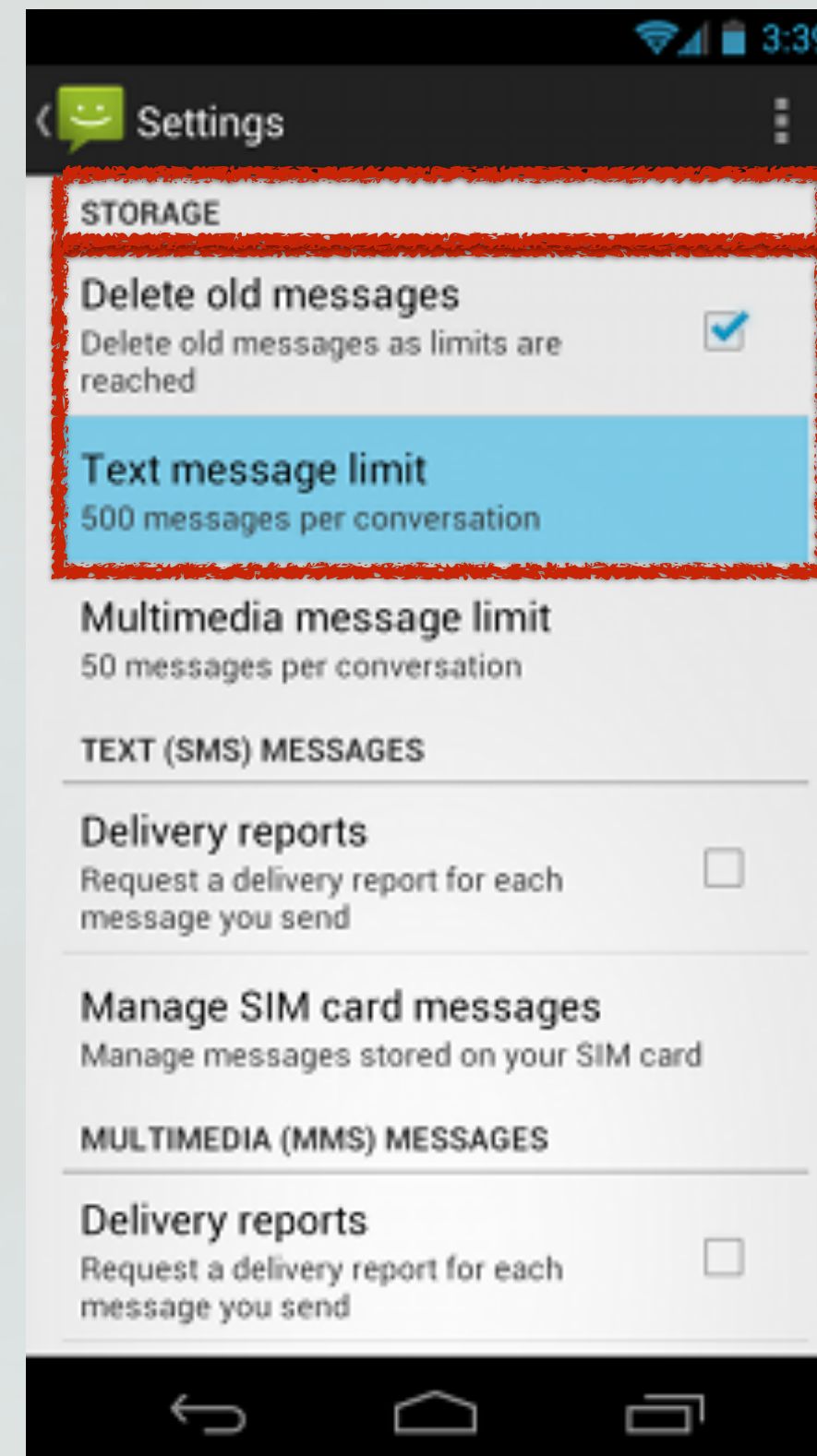
Not homogeneous with the other applications



BUT ...



Android is already providing necessary frameworks



Guidelines

Everything is configurable

 BUT you have to restrict configuration for your User

▶ **Otherwise your application will be too complex**

Guidelines according to API Level

< API 7	No groups
API 8 à API 10	Groups of two maximum
API 11 à API 15	Groups four maximum
API 16 +	Create a dedicated scree for groups of more than four elements

PreferenceActivity



Base class for building a hierarchy of preferences

- On small screen only displays titles
- On large screen, a two views are displayed side by side



Preferences are saved in SharedPreferences







Implement onBuildHeaders to separate headers and group of preferences

- A preference may have no group!

Simple Preferences (1/2)

A preference is composed of

-  a title: describing the preference
-  a summary: explaining what is the purpose of the preference
-  a key: that will be used by SharedPreferences
-  a default value: for the first launch of the application

Define a preference through an XML file

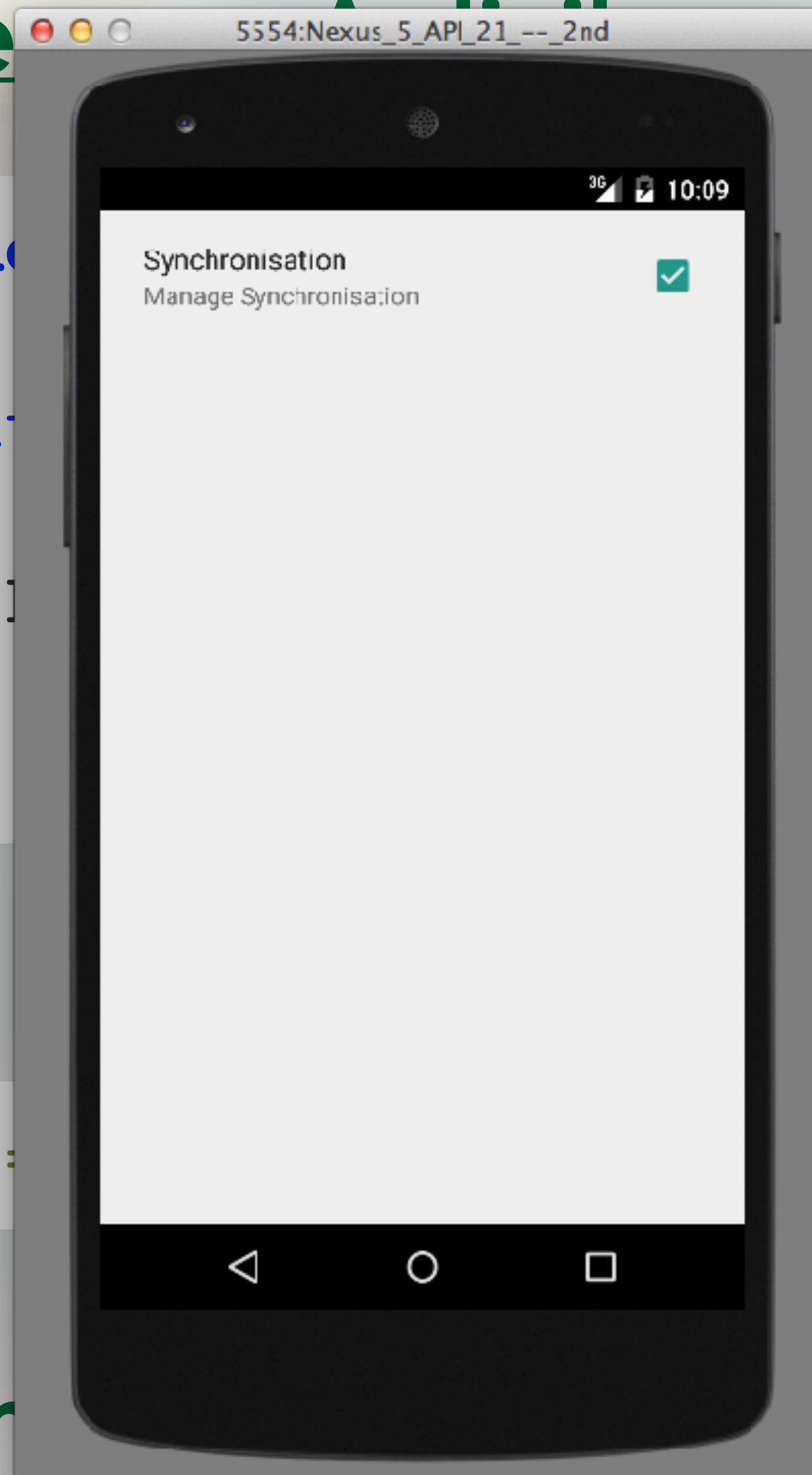
```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen
    xmlns:android="http://schemas.android.com/apk/res/android">
  <CheckBoxPreference
    android:key="@string/prefs_synchro"
    android:title="@string/synchro"
    android:summary="Manage Synchronisation"
    android:defaultValue="true" />
</PreferenceScreen>
```


Simple Preferences (2/2)



Inherits from Preference

```
public class SettingsAc
    @Override
    public void onCreate
        super.onCreate
        addPreferences
    }
}
```



```
PreferenceActivity{
    (onPreferenceState) {
    };
    (onPreference);
}
```



Declare the class

```
<activity android:name=
```

AndroidManifest.xml

```
></activity>
```



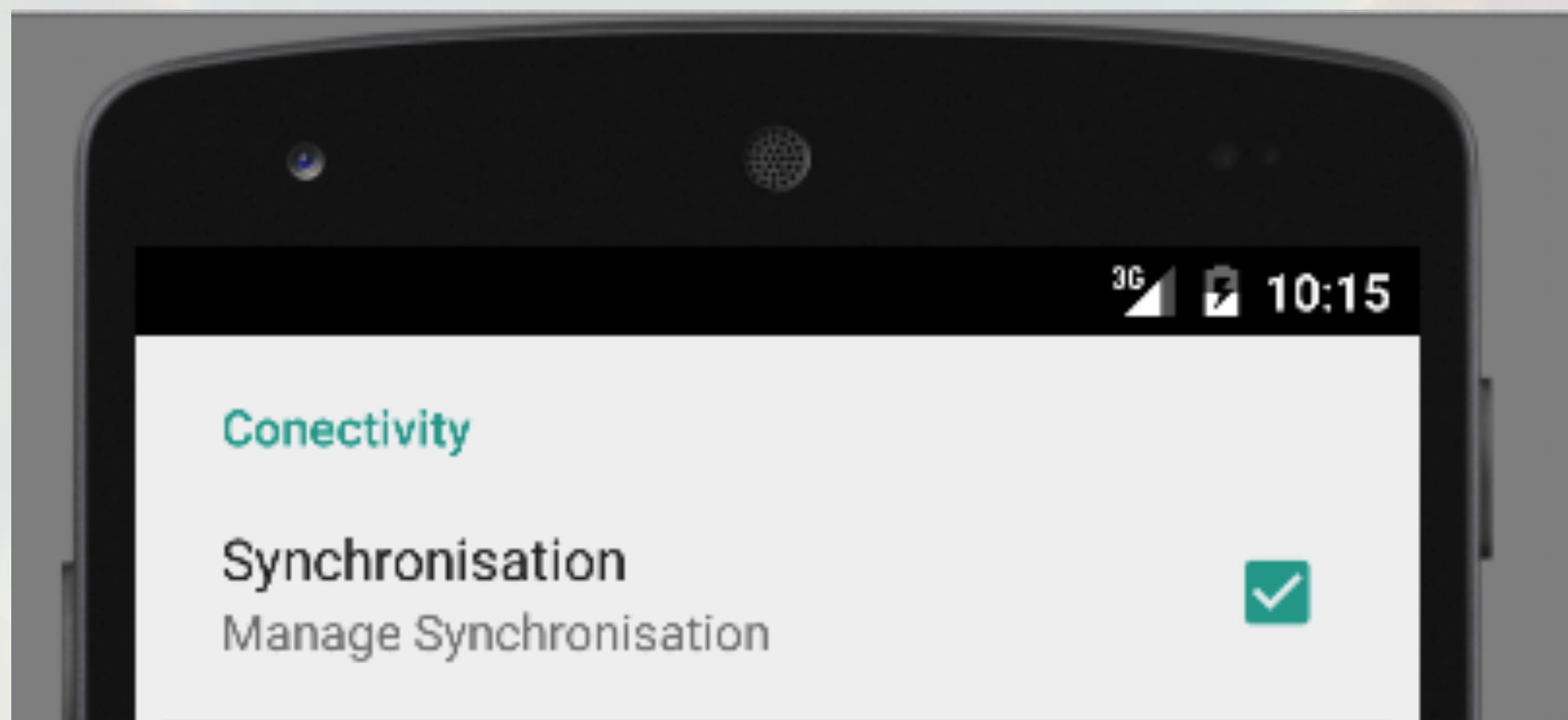
Run this class from

```
Intent i = new Intent(this, SettingsActivity.class);
startActivity(i);
```

Groups of Preferences

Define a group of Preferences

```
<PreferenceCategory
    android:key="pref_key_storage_settings"
    android:title="Conectivity">
    ...
</PreferenceCategory>
```



Run Intents from Preferences

Only specify it into the XML

```
Preference android:title="@string/prefs_web_page" >  
    <intent android:action="android.intent.action.VIEW"  
        android:data="http://www.example.com" />  
</Preference>
```

You can specify any kind of Intents

-  Just remember to declare the target activity into the AndroidManifest.xml

Manipulating Preferences (1/2)

Grab the value of a preference

```
SharedPreferences sharedPref =  
    PreferenceManager.getDefaultSharedPreferences(this);  
String syncConnPref =  
    sharedPref.getString(String.valueOf(R.string.synchro), "");
```

Register to watch modifications

 registerOnSharedPreferenceChangeListener:

▶ in the onStart method

 unregisterOnSharedPreferenceChangeListener:

▶ in the onStop method

 Implement OnSharedPreferenceChanged

Manipulating Preferences (2/2)



Callback when a Preference is changed

```
public void onSharedPreferenceChanged
    (SharedPreferences sharedPreferences, String key) {
    if (key.equals(R.string.synchro)) {
        Preference connectionPref = findPreference(key);
        // Set summary to be the user-description
        // for the selected value
        connectionPref.setSummary
            (sharedPreferences.getString(key, ""));
    }
}
```


Fragments of Preferences (1/2)



Define headers

```
<preference-headers
  xmlns:android="http://schemas.android.com/apk/res/android">
<header
  android:fragment="com.example.admin.preferenceapplication.SettingsA
  ctivity$Prefs1FragmentInner"
  android:title="Prefs 1"
  android:summary="Summary prefs 1" />
</preference-headers>
```



Define contents

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/
  android">
  <CheckBoxPreference
    android:defaultValue="true"
    android:key="@string/prefs_synchro2"
    android:summary="Manage Synchronisation"
    android:title="@string/synchro2" />
</PreferenceScreen>
```

Fragments of Preferences (2/2)

Implement the fragment

```
public static class Prefs1FragmentInner
    extends PreferenceFragment {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        addPreferencesFromResource(R.xml.fragmented_preferences);
    }
}
```

Modify the class extending PreferencesActivity to handle onBuildHeader and isValidFragment

```
public void onBuildHeaders(List<Header> target) {
    super.onBuildHeaders(target);
    loadHeadersFromResource(R.xml.preferences_headers, target);
}
protected boolean isValidFragment(String fragmentName) {
    return Prefs1FragmentInner.class
        .getName().equals(fragmentName);
}
```


Summary



Having the Android'Preferences look-and-feel is easy

- Just build an activity that inherits from PreferencesActivity
- The SharedPreferences component is used in background to save it



Split preferences to respect guidelines

- A title
- A content



DO NOT abuse of Preferences

- If an action tends to be frequently modified it may be useful to build a menu for that



