

Le backend : Webservices & API



Fil rouge



Intégration fil rouge

Demandez à votre formateur de vous partager le projet de base de l'application Fil rouge Android

Créez un repository sur votre compte gitHub et nommez le **TECHNIFUTUR-AND-FILROUGE**

Clonez le repository en local et ajoutez y les sources fournies par le formateur sur votre branche principale et commit + push

Tirez une branche develop depuis votre branche principale

Tirez une branche **feature/FR01** depuis la branche develop

Votre implémentation se fera sur cette branche

Une fois votre implémentation terminée, vous devrez "merge" votre branche sur **develop**

Comme le formateur est sympa, il va le faire avec vous pas à pas !





Intégration fil rouge

Nous allons reprendre le projet fil rouge pour:

- Intégrer un écran de login

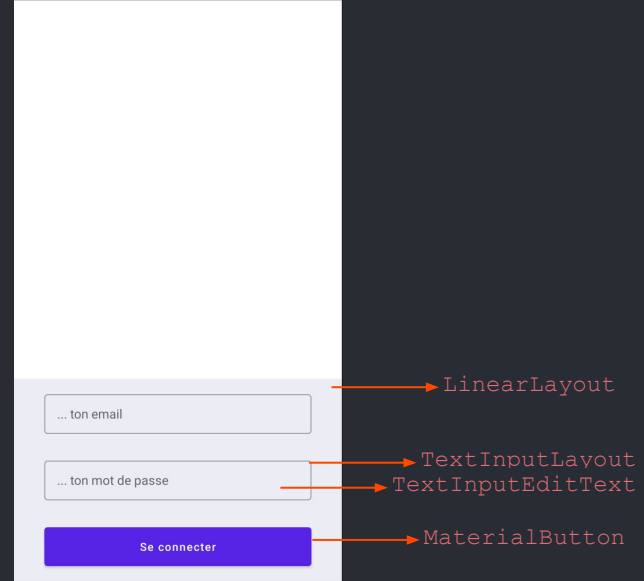
Pas de validation ou de navigation à implémenter pour l'instant

Nous allons uniquement préparer les layout pour l'étape suivante



Intégration fil rouge

Ouvrez la MainActivity et modifiez le layout afin d'arriver au résultat suivant





Intégration fil rouge

Demandez à votre formateur de vous partager le projet de base de l'application Fil rouge Android ou utilisez votre projet si vous êtes parvenus à réaliser l'étape précédente

Depuis develop, créez une branche feature/FR002

Votre implémentation se fera sur cette branche

Une fois votre implémentation terminée, vous devrez “merge” votre branche sur **develop**





Intégration fil rouge

Nous allons reprendre le projet fil rouge pour:

- Déplacer l'écran de login dans une nouvelle activité
- Ouvrir une activité home suite au login

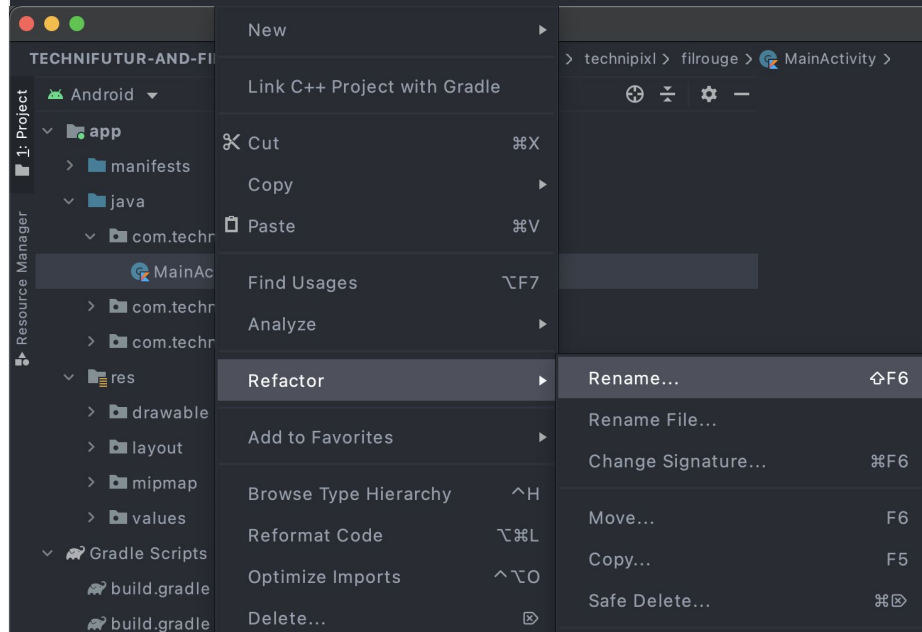


Intégration fil rouge

Première étape

Renommez la MainActivity en AuthenticationActivity

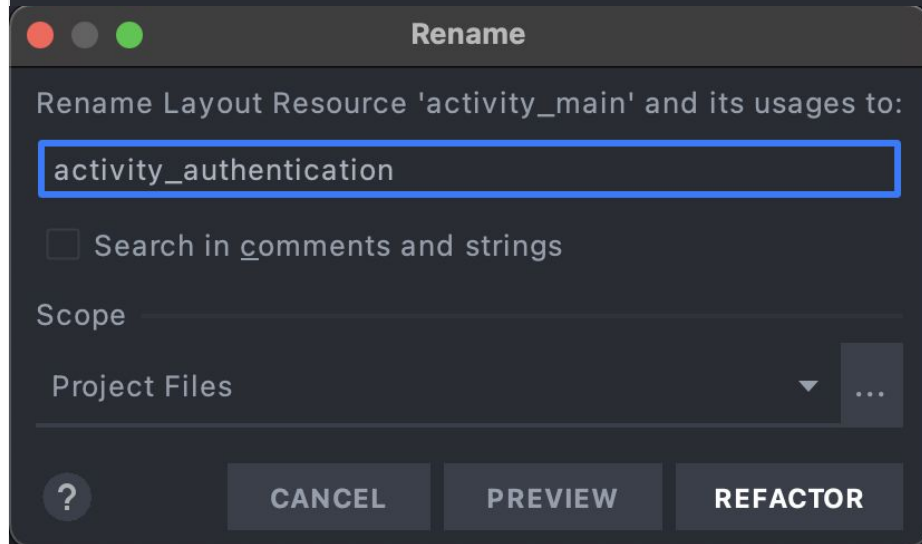
Pour ce faire, sélectionnez la MainActivity dans l'explorateur de projet, refactor + rename



Intégration fil rouge

Votre classe MainActivity et le fichier Manifest seront automatiquement mis à jour

Répétez l'opération pour le layout activity_main.xml

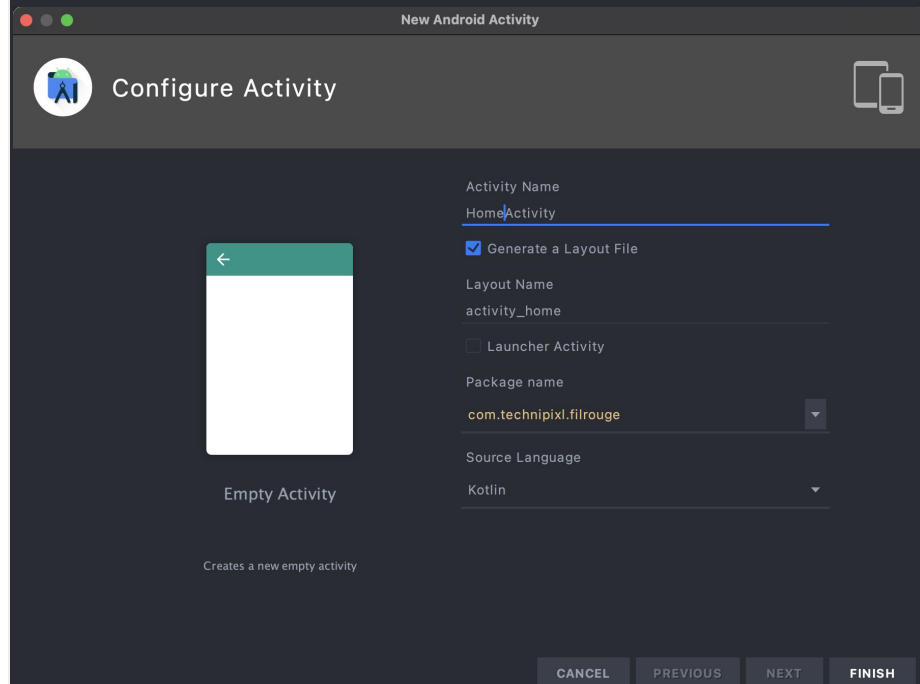


Intégration fil rouge

Ajoutez une nouvelle activité au projet et nommez la HomeActivity

À vous de jouer:

- Faites en sorte d'ouvrir la HomeActivity suite au login
- Uniquement si le login et le mot de passe ne sont pas vides



Intégration fil rouge

Demandez à votre formateur de vous partager le projet de base de l'application Fil rouge Android ou utilisez votre projet si vous êtes parvenus à réaliser l'étape précédente

https://github.com/technipixl/TECHNIFUTUR-AND-FILROUGE/releases/tag/home_navigation

Depuis develop, créez une branche feature/FR003

Votre implémentation se fera sur cette branche

Une fois votre implémentation terminée, vous devrez “merge” votre branche sur **develop**

Demandez aussi à votre formateur de vous partager les fichiers nécessaires à la configuration de votre menu



Intégration fil rouge

Implémenter un graphe de navigation:

- Ajoutez une ressource de type navigation et nommez là nav_graph
- Ajoutez votre “host” de navigation dans le layout de la HomeActivity

```
<androidx.fragment.app.FragmentContainerView
    android:id="@+id/nav_host_fragment"
    android:name="androidx.navigation.fragment.NavHostFragment"
    android:layout_width="0dp"
    android:layout_height="0dp"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:defaultNavHost="true"
    app:navGraph="@navigation/nav_graph" />
```

Intégration fil rouge

- Ouvrez votre nav_graph
- Ajoutez une nouvelle destination et nommez là HomeFragment

Hosts

[Create new destination](#)

No NavHostF

This nav g
referencNavHostFragm
order to be

Using Navigati



placeholder

Empty destination



activity_authentication

Activity



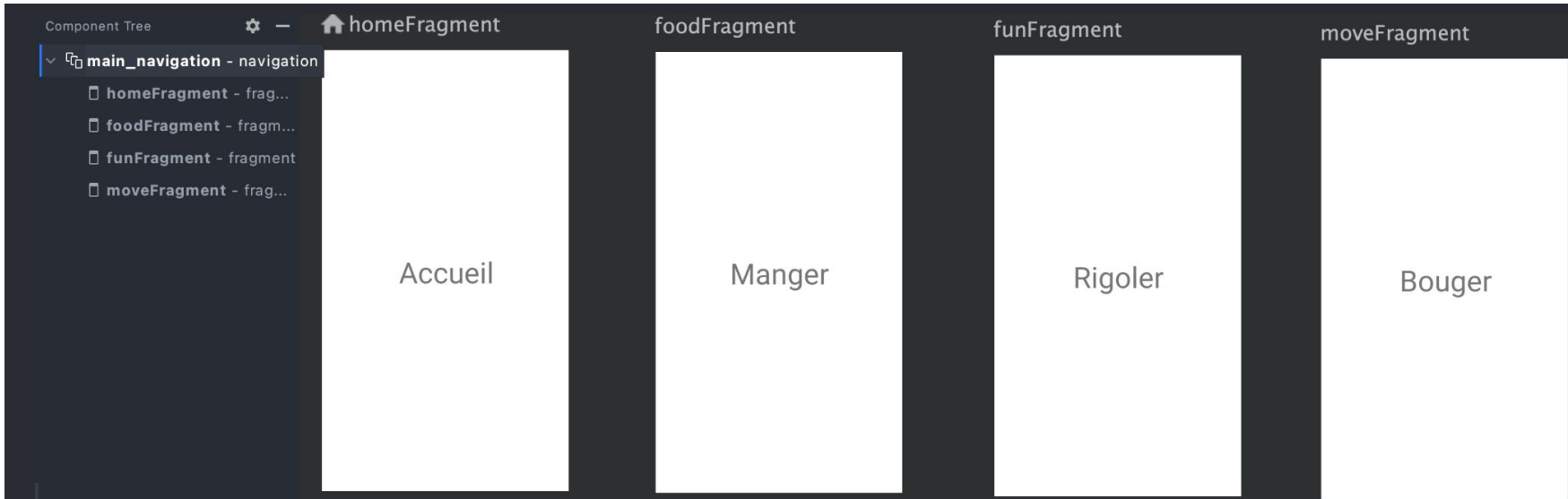
activity_home

Activity

Intégration fil rouge

Implémenter un graphe de navigation:

- Répétez l'opération et ajoutez FoodFragment, FunFragment, MoveFragment
- Modifiez les layout de vos fragments pour afficher leur nom dans une TextView centrée





Intégration fil rouge

- Ajoutez une ressource de type menu et nommez la nav_menu
- Ajoutez les drawable à votre projet
- Définissez votre menu avec pour identifiants d'éléments, les identifiants de vos fragments

```
<?xml version="1.0" encoding="utf-8"?>
<menu
  xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto">
  <item
    android:id="@+id/homeFragment"
    android:title="@string/home_menu_title"
    android:icon="@drawable/ico_house_white"
    app:showAsAction="ifRoom|withText"/>

  <item
    android:id="@+id/foodFragment"
    android:title="@string/food_menu_title"
    android:icon="@drawable/ico_pizza_white"
    app:showAsAction="ifRoom|withText"/>

  <item
    android:id="@+id/funFragment"
    android:title="@string/fun_menu_title"
    android:icon="@drawable/ico_laugh_white"
    app:showAsAction="ifRoom|withText"/>

  <item
    android:id="@+id/moveFragment"
    android:title="@string/move_menu_title"
    android:icon="@drawable/ico_car_white"
    app:showAsAction="ifRoom|withText"/>
</menu>
```



Intégration fil rouge

Implémenter une bottom navigation:

- Ajouter une bottom navigation view à la HomeActivity
- Placez la en base de l'écran
- Modifier aussi la contrainte bottom de votre FragmentContainerView pour se caler sur le top de votre BottomNavigationView

```
<com.google.android.material.bottomnavigation.BottomNavigationView  
    android:id="@+id/bottomNavigationView"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:menu="@menu/nav_menu"  
    app:labelVisibilityMode="labeled"/>
```


Intégration fil rouge

Implémenter une bottom navigation:

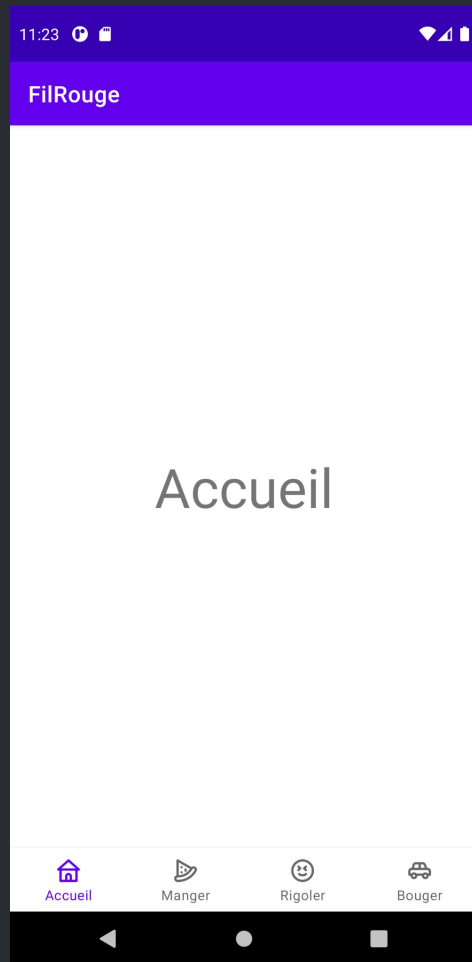
- Intégrer la bottom navigation view au nav controller
- Dans la méthode `onPostCreate` de votre `HomeActivity`, liez la bottom navigation à votre nav controller

```
override fun onPostCreate(savedInstanceState: Bundle?) {  
    super.onPostCreate(savedInstanceState)  
    setupBottomNavigation()  
}  
  
private fun setupBottomNavigation() {  
    val navController = Navigation.findNavController(this, binding.navHostFragment.id)  
    NavigationUI.setupWithNavController(binding.bottomNavigationView, navController)  
}
```



Intégration fil rouge

Résultat final





Intégration fil rouge

Demandez à votre formateur de vous partager le projet de base de l'application Fil rouge Android ou utilisez votre projet si vous êtes parvenus à réaliser l'étape précédente

Depuis develop, créez une branche feature/FR004

Votre implémentation se fera sur cette branche

Une fois votre implémentation terminée, vous devrez “merge” votre branche sur **develop**

Demandez aussi à votre formateur de vous partager les fichiers drawable nécessaires à la mise en place de votre écran





Intégration fil rouge

Nous allons reprendre le projet fil rouge pour:

- Améliorer l'expérience utilisateur de l'écran de login



Intégration fil rouge

Améliorer l'expérience utilisateur de l'écran de login:

- Ajouter les éléments graphiques (background, styles, fonts, info bulle informative...)
- Gérer l'affichage du clavier
 - La zone de saisie active doit rester visible
- Gérer l'affichage des cas d'erreur
 - Le bouton "Se connecter" doit s'activer uniquement si les règles de validation sont respectées
 - La couleur du texte et le pictogramme affiché doit varier selon la validation des règles
- Gérer la validation du formulaire
 - L'email doit respecter un regex de type email

```
val EMAIL_REGEX: Pattern = android.util.Patterns.EMAIL_ADDRESS
```

```
val PASSWORD_REGEX: Pattern = Pattern.compile("(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-9])(?=.*?[#?!@$%^&*~]).{8,}$")  
val isValid = !password.isNullOrEmpty() && Validators.PASSWORD_REGEX.matcher(password).matches()
```

Intégration fil rouge

Rappel sur les composants :

- TextInputLayout: gestion des erreurs
- TextInputEditText: encodage du texte

```
<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/emailInputLayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginHorizontal="@dimen/input_horizontal_margins"
    android:layout_marginTop="39dp"
    android:layout_marginBottom="@dimen/input_vertical_margins"
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    app:errorIconDrawable="@drawable/ico_cross_rose"
    app:startIconDrawable="@drawable/ico_profil_blue"
    android:background="@drawable/input_rounded_background"
    app:hintEnabled="false">
    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/emailEditText"
        android:layout_width="match_parent"
        android:layout_height="@dimen/input_height"
        android:inputType="textEmailAddress"
        android:hint="@string/email_hint"
        android:background="@android:color/transparent"
        android:textSize="@dimen/input_text_size"/>
</com.google.android.material.textfield.TextInputLayout>
```



Intégration fil rouge

Comment activer et désactiver le message d'erreur sur un composant **TextInputLayout** et modifier la couleur du texte:

```
private fun toggleEmailValidity(isValid: Boolean) {  
    if(isValid) {  
        binding.emailInputLayout.isErrorEnabled = false  
        binding.emailEditText.setTextColor(ResourcesCompat.getColor(resources, R.color.text_input_valid_text_color,  
theme))  
    } else {  
        binding.emailInputLayout.error = getString(R.string.email_error_hint)  
        binding.emailInputLayout.isErrorEnabled = true  
        binding.emailEditText.setTextColor(ResourcesCompat.getColor(resources, R.color.black, theme))  
    }  
}
```

Intégration fil rouge

Comment définir un background de vue personnalisé:

- Créez une ressource de type Drawable resource file (ex: input_rounded_background.xml)
- Définissez son aspect (ex: ci-dessous pour un rectangle avec bords arrondis)
- Définissez le background de votre vue: **android:background="@drawable/input_rounded_background"**

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle"
    android:padding="10dp">

    <solid android:color="@color/white" />
    <corners
        android:bottomRightRadius="25dp"
        android:bottomLeftRadius="25dp"
        android:topLeftRadius="25dp"
        android:topRightRadius="25dp" />
</shape>
```


Intégration fil rouge

Pour un fond avec gradient:

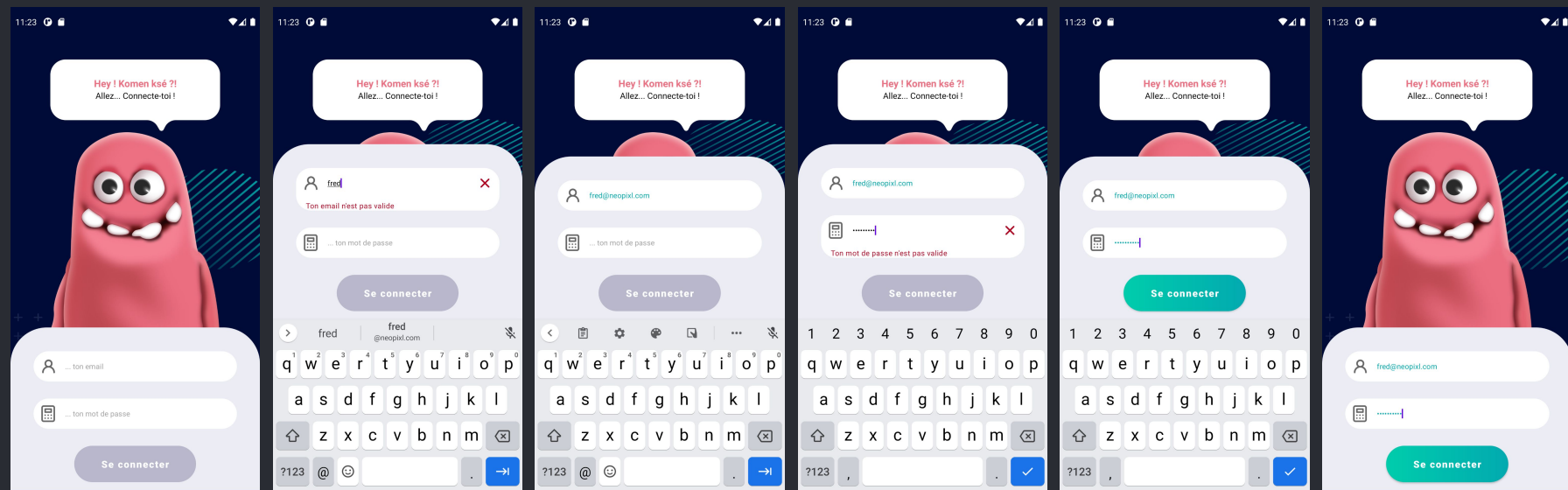
- Cette fois nous utiliserons un élément de type **gradient** à la place d'un élément de type **solid**

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle"
    android:padding="10dp">

    <gradient android:startColor="@color/green_gradient_start" android:endColor="@color/green_gradient_end"
    android:angle="0" />
    <corners
        android:bottomRightRadius="@dimen/button_corner_radius"
        android:bottomLeftRadius="@dimen/button_corner_radius"
        android:topLeftRadius="@dimen/button_corner_radius"
        android:topRightRadius="@dimen/button_corner_radius" />
</shape>
```

Intégration fil rouge

Voici le rendu final attendu





Restons en contact

neopixl.

A SMILE GROUP COMPANY

115A, Rue Emile Mark
L-4620 Differdange

(+352) 26 58 06 03
contact@neopixl.com