

## Série de TP

### Remarques :

1. L'évaluation se fait chaque séance, la note finale sera la moyenne de toutes les notes.
2. Les étudiants sont appelés à avancer le maximum chez eux.
3. De préférences les TP se font en monôme.
4. Ne ramenez pas des projets d'internet, on peut les trouver, nous aussi on a Google
5. Les TP se font en Android native. Les multi-plateformes sont autorisées (Flutter, Xamarine, React Native).

#### 1. **TP1 : First App**

- a. Créez un projet sous Android Studio, Nommer le projet comme suit : **VotreNomPrenomGrp**
- b. Ajoutez un **TextView** et mettez dedans votre nom et prénom.
- c. Modifier la taille du texte a **30**
- d. Modifier la couleur du texte en « **Bleu** ».
- e. Arranger le d'une façon qu'il se positionne en centre de l'écran.

Lancez l'exécution de l'application.

#### 2. **TP2 : USER INTERFACE**

Il est demandé de réaliser un écran de login. Il contient deux **EditText** un pour le **user\_name** et l'autre pour le **password**. Nommer les **nameTxt** et **pwdTxt**, Il contient deux **Buttons** , un pour valider l'action et l'autre pour l'annulation, nommer les **okBtn** et **cancelBtn**.

Faites un **Toast** pour afficher l'information saisi par l'user.

Partie 2 :

Ajouter un evennement au bouton okBtn, récupérer le user name saisi et le mot de passe, tester si le user name est « **test** » et le mot de passe et « **password** », si c'est le cas afficher login réussi , sinon afficher login incorrect merci de ressaisir les informations de login

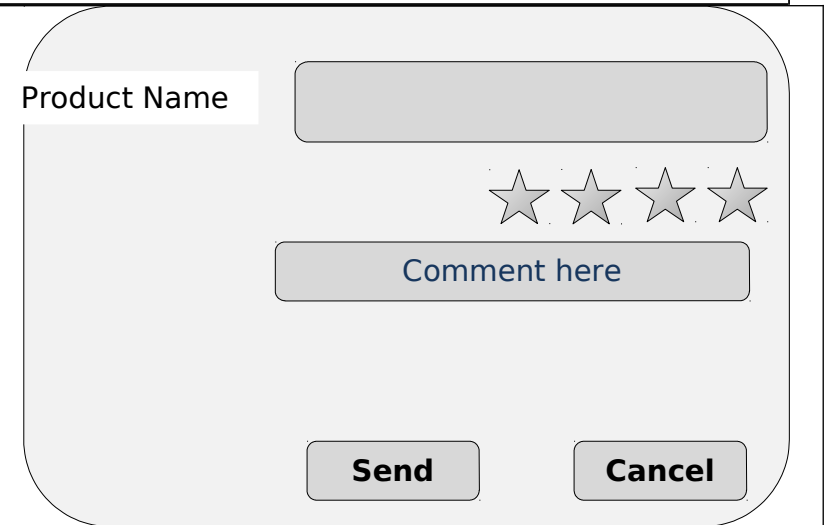
#### 3. **TP3: Calculatrice en trois valeurs**

Il est demandé de réaliser une application qui permet à l'utilisateur de saisir trois valeurs numériques, après l'application offre a l'utilisateur les opérations suivant :

1. Calculer la moyenne et l'afficher dans une zone textview
2. Calculer la valeur maximale et l'afficher dans la zone textview
3. Calculer la valeur minimale.
4. Permettre un bouton qui permet d'effacer les troix zones de saisie.

4. TP4 : **USER INTERFACE**

- a. Il s'agit de créer une interface utilisateur qui permet de saisir un avis concernant un **produit** comme schématisé dans la figure.
  - i. Le nom de produit est **indispensable**
  - ii. Le nombre de caractères du commentaire doit être **> 20** et **< 60**.
- b. Si les informations saisi ne sont pas valide un message d'erreur avec indication de l'erreur est affiché.
- c. Si les données sont valides un message «Merci pour avoir donner votre avis » est affiché



The image shows a user interface for submitting a product review. It features a rounded rectangular container with a light gray background. At the top left, the text 'Product Name' is followed by a text input field. Below this, there are four gray star icons for rating. Underneath the stars is a text input field with the placeholder text 'Comment here'. At the bottom of the container, there are two buttons: 'Send' and 'Cancel'.

Interface de saisi d'un avis concernant un produit.

5. **TP5 Choix de meilleur langage de programmation :**

Créer une application qui permet à l'utilisateur de choisir parmi les langages de programmations suivant : **Java, Javascript, Dart, Python, C#, C++.**

6. **TP6 THE GAME**

- a. Il s'agit de réaliser le jeux **sudoku** qui fonctionne comme suit : le joueur doit remplir les cellules vides par un chiffre, ce dernier ne peut **figurer** qu'une seule fois dans la ligne et une seule fois dans la colonne. Les cellules voisines doivent avoir des chiffres consécutifs. Comme décrit dans la figure. Dans l'exemple on ne peut utiliser que **1,2,3,4**
- b. Utilisez des **TextView** pour les chiffres fixes et des **EditText** pour les cellules vides.
- c. Le jeu se compose d'une seule **Activity**, avec deux **Button** : un pour démarrer le jeu l'autre annule la partie. Un **Textview** est affiche en haut qui affiche le nombre de partie terminées.
- d. **Optionnel** :

	3		1
1		3	2
3		1	
	1		3

- i. Ajouter un **Timer** qui affiche le temps maximum pour la partie pour être résolue.
- ii. Afficher un **Button** « level » qui permet de générer un sudoku avec une matrice plus grande.
- iii. Le jeu indique en couleur rouge le chiffre qui n'est pas correcte.

### **7. TP 6 Livre de recette**

On souhaite réaliser une application mobile qui contient des recettes de cuisine. L'application utilise une base de données local **SQLITE**, le schéma de la base de données est très simple est ne contient qu'une seule table :

**Recette( ID INTEGER, TITRE VARCHAR(30), ETAPES VARCHAR(300), keywords VARCHAR(30)).**

L'application est composée de 5 **page** :

- Page d'accueil, elle affiche un menu principal.
- Page recherche qui permet de chercher si une recette existe dans la base de donnée, si elle existe elle sera affichée en détails dans une page.
- Page Ajouter qui permet d'ajouter une recette
- Page supprimer qui permet de supprimer une recette.

#### **Remarque :**

Une classe Java sera fournie par le chargé de cours, elle contient les traitements base de données (insertion, recherche, suppression etc), attention elle n'est pas complète vous devez la corriger et l'intégrer dans votre projet.