

Évaluation 2 : Développement d'une application Flutter

Module : Développement natif, avec Dart et Flutter

Promotion : M2 Dev 2023/2024

Version : 1

Date : 25/10/23

Noté sur 20, coefficient 1

Comment rendre votre travail

Envoyer votre travail dans **un e-mail** à l'adresse suivante:
paul.schuhmacher.ext@eduservices.org, **ayant le sujet suivant** :

rendu - dev natif - x_abc

où **x** est la première lettre de votre nom et **abc** de votre prénom (j'enverrai donc un e-mail avec le sujet **rendu - dev natif - s_paul**).

Dans l'e-mail :

- **Mettre le lien de votre dépôt Github/Gitlab qui héberge votre projet.**

Grille d'évaluation

Le travail rendu sera noté sur 20 selon les critères suivants :

- **Documentation du projet : 7 points.** Le projet doit être versionné correctement avec git et hébergé sur Github/Gitlab (ou autre). Le dépôt doit contenir *a minima* un **README** avec les instructions pour installer, lancer votre application et accéder à la documentation. Ajouter au **README** une section **Références/Ressources** qui liste les ressources qui vous ont servi durant le développement de l'application. Penser à ajouter une courte description de chaque ressource. Ajouter également les sections **Difficultés**, où vous pouvez documenter les difficultés rencontrées, et **Choix de design/Implémentation**, où vous pouvez discuter de vos choix de conception (en plus des commentaires éventuels dans le code). L'historique des commits sera également regardé.
- **Maintenabilité du code : 4 points.** Le code doit être *simple à lire et simple à modifier*. Seront pris en compte : le formatage du code et le respect de standards, les signatures de fonction (nommage, interfaces claires), le choix des Widgets, le découpage du code en

module (fichiers), l'utilisation des codeblocks de flutter pour générer la documentation, etc.

- **Implémentation : 9 points.** Votre projet doit implémenter les spécifications demandées (suivre les consignes). Vous pouvez ajouter des fonctionnalités *en plus* si vous le souhaitez, une fois les spécifications demandées implémentées. Penser à les documenter dans votre **README**.

Projet

Ce projet s'appuie sur les connaissances vues en cours (Stateful et Stateless Widget, Navigation, gestion d'état, composition de widgets).

Développer une application Android de [clicker game ou incremental game](#), dans la tradition du célèbre [Universal Paperclips](#) créée par [Frank Lantz](#) en 2017.

Le jeu doit permettre de gérer un stock de ressources à collecter et dépenser. Voici la liste de ressources de départ :

- Bois
- Minerai de fer brut
- Minerai de cuivre brut

La personne qui joue doit pouvoir cliquer sur chaque ressource pour en récolter 1 unité. Les ressources récoltées permettent de débloquent des recettes et de nouveaux éléments de gameplay pour progresser dans le jeu, et ainsi de suite.

1. Créer une première page d'accueil dont le titre est *Ressources*. Le jeu démarre tout de suite. Utiliser un widget **GridView** pour présenter les ressources disponibles. La **GridView** doit contenir 4 colonnes, peu importe la taille de l'écran. Chaque ressource est un widget **Ressource** représenté par un rectangle de couleur sur la **GridView** (voir le tableau des spécifications pour les détails)
2. Ajouter un bouton au widget **Ressource** ayant le label **Miner**. Lorsque ce bouton est cliqué, une unité de ressource doit être récoltée. Afficher la quantité totale de ressource récoltée sur chaque ressource.
3. Ajouter un bouton dans l'**AppBar** de la route *Ressources* qui permet de *naviguer* vers une nouvelle Route *Recettes*. Utiliser la méthode de navigation que vous souhaitez. Utiliser une icône pertinente parmi celles proposées par [Google Font](#). Cette route doit afficher toutes les recettes du jeu sous forme de liste dans une **ListView**. Chaque recette doit afficher le nom de la recette, sa description et son coût en ressources. Pour chaque recette, ajouter un bouton *Produire* pour produire l'objet. Un objet produit est ajouté à l'inventaire.
4. Le bouton *Produire* ne doit pas fonctionner (ne rien faire) si les ressources ne sont pas suffisantes pour produire l'objet. *Bonus: donner un "look" disabled au bouton si la recette n'est pas disponible*

5. Ajouter un bouton dans l'**AppBar** de la route *Ressources* qui permet de *naviguer* vers une nouvelle Route *Inventaire*. Cette route doit afficher tous les items produits sur la route *Recettes* dans une **ListView**. Chaque item doit afficher son nom et la quantité produite.
6. Lorsque le joueur a fabriqué 1000 lingots de fer et 1000 lingots de cuivre, la nouvelle ressource à récolter *Charbon* est affichée et disponible sur la route *Ressources*.
7. Ajouter un bouton dans l'**AppBar** de la route *Inventaire* qui permet de trier l'inventaire par nom. Ajouter un bouton dans l'**AppBar** de la route *Inventaire* qui permet de trier l'inventaire par quantité.
8. Ajouter un nouvel élément de gameplay de votre choix : une nouvelle page, une nouvelle action, une nouvelle interaction ou une nouvelle recette. **Cet élément de gameplay doit inclure au moins un nouveau widget** dans l'application. Documenter le dans votre README. Rappelez-vous qu'un *idle game* fonctionne sur l'addiction, le plaisir de regarder les chiffres monter et le temps long.
9. **Bonus** : un Mineur ou une Fonderie sont des bâtiments qui permettent d'automatiser la production. C'est un élément de gameplay essentiel d'un clicker game, où le jeu doit pouvoir avancer sans notre intervention (c'est pourquoi on appelle ces jeux idle game, jeux d'attente/passif). Lorsqu'un Mineur est fabriqué et dans l'inventaire, il s'applique automatiquement aux ressources métalliques (fer et cuivre). Un mineur extrait **3 items/seconde**. Lorsqu'une fonderie est dans l'inventaire, elle transforme automatiquement les minerai brut de fer et de cuivre en lingots. Une fonderie transforme **1 minerai brut/seconde**. On ne peut pas construire plus de mineurs qu'il n'y a de type de ressources (ici 2). Implémenter les recettes Mineur et Fonderie dans le jeu.

Si un effort est fait sur la présentation (layouts), l'aspect visuel des widgets, il sera pris en compte également (bonus).

Si vous ne répondez pas à une question bonus vous n'êtes pas pénalisé-e. Y répondre apporte des points supplémentaires.

Conseils

- Réfléchissez en amont à l'organisation de votre code, aux abstractions utiles, etc.
- Si vous bloquez, passer à une autre question.
- Utiliser des *placeholders* (fausses valeurs en dur) pour simuler des états dans un premier temps et fabriquer l'assemblage de widgets pour chaque Route (Page). Intégrez ensuite les états réels de l'application.
- Lisez la documentation pour apprendre à utiliser les widgets
- Inspirez vous des démos fournies
- Prenez votre temps
- *Si vous avez le temps ou l'envie*, améliorez le gameplay et proposer de nouveaux éléments ou allez plus loin (nouvelles recettes, nouvelles ressources, etc.). N'oubliez pas de les documenter dans votre README. Cela sera pris en compte.
- Have fun !

Données

Ressource	Couleur	Description
Bois	#967969	Du bois brut
Minerai de fer	#ced4da	Du minerai de fer brut
Minerai de cuivre	#d9480f	Du minerai de cuivre brut
Charbon	#000	Du minerai de charbon

Recette	Coût	Gameplay	Type	Description
Hache	2 Bois, 2 Tige en métal	Récolter le bois 3 par 3 (3 par click)	Outil	Un outil utile
Pioche	2 Bois, 3 Tige en métal	Récolter les minerais 5 par 5 (5 par click)	Outil	Un outil utile
Lingot de fer	1 Minerai de fer	Débloque d’autres recettes	Matériau	Un lingot de fer pur
Plaque de fer	3 Minerai de fer pour 2 plaques de métal	Débloque d’autres recettes	Matériau	Une plaque de fer pour la construction
Lingot de cuivre	1 Minerai de cuivre	Débloque d’autres recettes	Matériau	Un lingot de cuivre pur
Tige en métal	1 Lingot de fer	Matériau	Débloque d’autres	Une tige de métal

Fil électrique	1 Lingot de cuivre	Composant	Débloque d'autres recettes	Un fil électrique pour fabriquer des composants électroniques
Mineur	10 Plaque de fer, 5 Fil électrique	Bâtiment	Permet d'extraire automatiquement du minerai de fer ou cuivre	Un bâtiment qui permet d'automatiser le minage
Fonderie	5 Fil électrique, 8 Tige en métal	Bâtiment	Permet de transformer automatiquement du minerai de fer ou cuivre en lingot de fer ou cuivre	Un bâtiment qui permet d'automatiser la production.

Toute similitude avec un jeu ou plusieurs jeux existants n'est pas intentionnelle.