

TP - Mona Lisa

UI - Responsive - Initiation à la gestion d'état - Refactoring

Veuillez consulter les consignes générales de réalisation et de rendu fournies en amont.

Objectifs

- Prise en main de Flutter,
- Découverte de la mise en page à l'aide de widgets,
- Gestion des interactions de l'utilisateur,
- Gestion de l'état applicatif (Stateless / Stateful),
- Emploi d'assets,
- Affichage conditionnel.

Etape 1 : Initialisation

(cf. structure de l'application en annexe).

1. Créez un nouveau projet Flutter avec la commande
`flutter create monalisa`
2. Supprimez le code généré par défaut dans le fichier ***./lib/main.dart*** et dans le fichier ***./test/widget_test.dart***
3. Importez le package ***flutter/material.dart*** à la première ligne du fichier ***./lib/main.dart***
4. Créez une classe nommée ***MuseumApp*** héritant de ***StatelessWidget***
5. Faites en sorte que la méthode ***build*** de ce widget retourne un widget de type ***MaterialApp***.
Affectez à l'attribut ***title*** du widget ***MaterialApp***, la valeur "***Museum***".
Pour l'attribut ***theme*** du widget ***MaterialApp***, renseignez le ***ThemeData*** configuré avec la couleur primaire de votre choix (ex: *Brown*).
6. Créez une nouvelle classe nommée ***Artwork*** héritant de ***StatelessWidget***. Employez un widget ***Scaffold*** dans la méthode ***build*** de ce widget. Intégrer les widgets nécessaires à Scaffold pour obtenir un rendu graphique (ex : *appBar*, *body*, *floatingActionButton*...).
7. Dans le widget ***MuseumApp***, employez une instance du widget ***Artwork*** en tant que valeur de l'attribut ***home*** du widget ***MaterialApp***
8. Créez une méthode nommée ***main*** préfixée par le retour de type ***void***
Au sein de la méthode ***main***, exécutez la méthode ***runApp*** avec en paramètre une instance ***MuseumApp***
9. Afin de vérifier son fonctionnement, exécutez l'application (dans votre navigateur web Chrome ou Edge, ou dans un émulateur mobile ou encore dans un device connecté). Employez la commande ***flutter run***

Etape 2 : Scaffold

1. Dans le widget **Artwork**, employez un widget de type **Scaffold** en tant que widget racine retourné par la méthode **build**.
2. Pour l'attribut **appBar** du widget **Scaffold**, utilisez un widget de type **Text** pour afficher la mention "**Museum**"

Etape 3 : Assets

1. Créez un dossier **./images** et placez le fichier **Mona_Lisa.jpg** (fourni).
2. Adaptez le fichier **./pubspec.yaml** afin d'intégrer le fichier **Mona_Lisa.jpg** au build de l'application (attention à l'indentation du fichier yaml).
3. Téléchargez la police **Merriweather** depuis Google Fonts (<https://fonts.google.com/specimen/Merriweather>). Dézippez l'archive et placez les fichiers **.ttf** dans le dossier **./fonts/merriweather**

Etape 4 : UI

(cf. captures d'écran en annexe).

1. Dans le widget **Artwork**, affichez l'image **Mona_Lisa.jpg**. Centrez l'image horizontalement.
2. Sous l'image, affichez le texte "**Mona Lisa**" avec la police **Merriweather**, **fontSize** à 30, en couleur **Brown**.
3. Sous le texte "**Mona Lisa**", affichez un second texte contenant la mention "**Léonard De Vinci**", avec la police **Merriweather**, **fontSize** à 15, **fontWeight** à la valeur **bold**, en couleur **Brown**.
4. Superposer à l'image, un widget **Icon** avec le symbole nommé **favorite**, **fontSize** à 100, couleur **White**, avec une **opacité** de 75%, centré horizontalement et

verticalement.

5. Sous le texte "**Léonard De Vinci**", affichez deux **IconButton**, un avec l'icône **article** et l'autre, avec l'icône **favorite**.

Etape 5 : StatefulWidget

1. Convertissez le widget **Artwork** en **StatefulWidget**, afin de disposer d'un état applicatif (state) permettant de mettre à jour l'interface (UI) en fonction des interactions de l'utilisateur et de l'évolution des variables d'état.
2. Créez une variable d'état **_isFavorite** de type **bool**, initiée à **false**
3. Lorsque l'utilisateur clique sur l'**IconButton** de type **favorite** située en-dessous de l'image, alternez la valeur (**true / false**) de la variable d'état **_isFavorite**
4. Faites en sorte que la couleur de *l'icône superposée à l'image, passe de White à Red et que son opacité passe de 75 à 100%*, lorsque **_isFavorite** a la valeur **true**
5. Alternez également la couleur de l'IconButton **favorite** placée sous l'image de **Brown** à **Red**, lorsque **_isFavorite** passe de **false** à **true**
6. A l'aide d'un widget **Snackbar**, affichez une notification (ex: "Oeuvre ajoutée à vos favoris") lorsque l'utilisateur clic sur l'**IconButton favorite** et que la valeur de **_isFavorite** est à **true**

7. Au clic sur l'**IconButton** de type **article**, affichez / masquez le texte suivant à l'aide d'une variable d'état, nommée **_showDescription** de type **bool** :

La Joconde, ou Portrait de Mona Lisa, est un tableau de l'artiste Léonard de Vinci, réalisé entre 1503 et 1506 ou entre 1513 et 1516, et peut-être jusqu'à 1517 (l'artiste étant mort le 2 mai 1519), qui représente un portrait mi-corps, probablement celui de la Florentine Lisa Gherardini, épouse de Francesco del Giocondo. Acquise par François Ier, cette peinture à l'huile sur panneau de bois de peuplier de 77 × 53 cm est exposée au musée du Louvre à Paris. La Joconde est l'un des rares tableaux attribués de façon certaine à Léonard de Vinci.

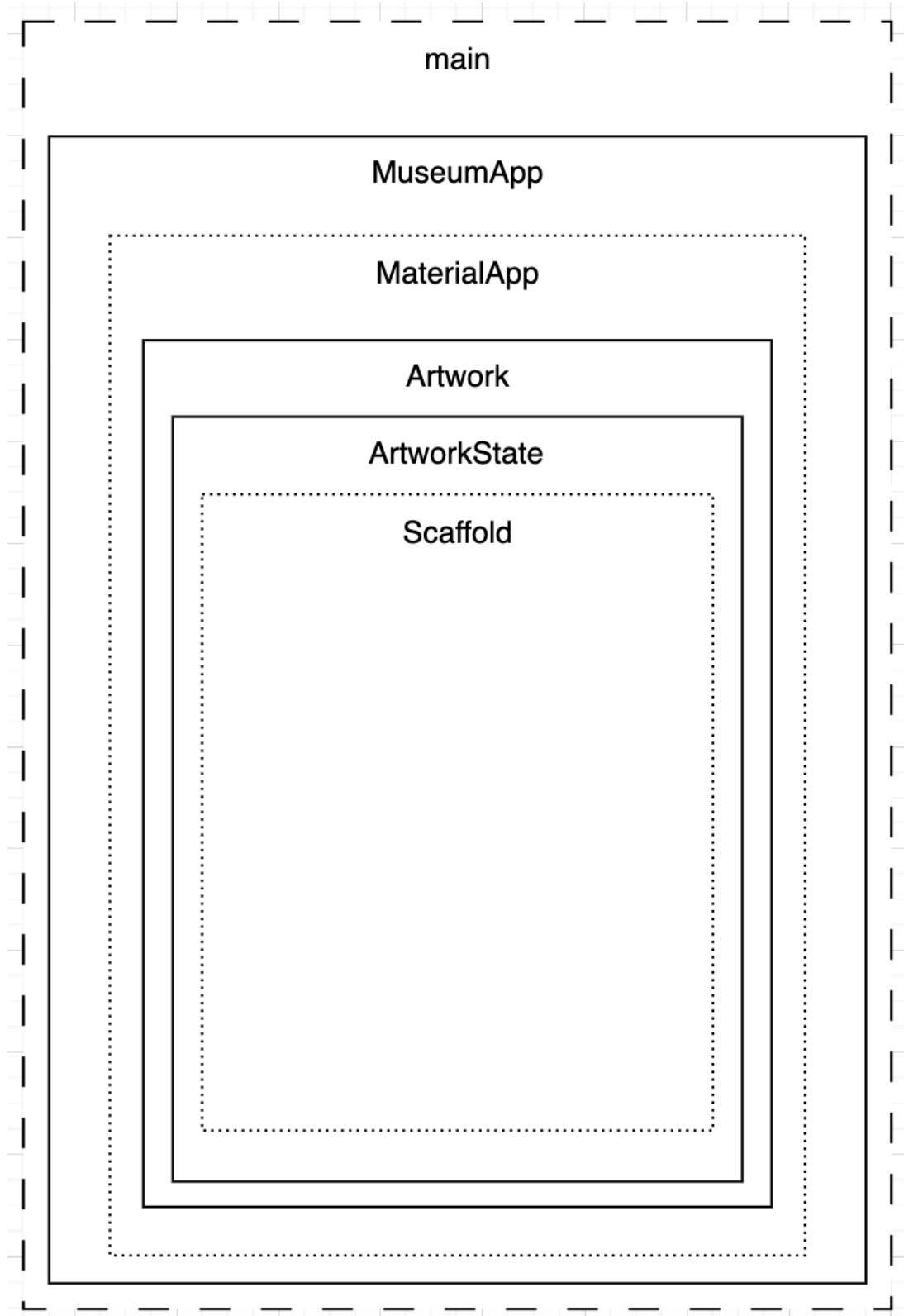
La Joconde est devenue un tableau éminemment célèbre car, depuis sa réalisation, nombreux d'artistes l'ont pris comme référence. À l'époque romantique, les artistes ont été fascinés par ce tableau et ont contribué à développer le mythe qui l'entoure, en faisant de ce tableau l'une des œuvres d'art les plus célèbres du monde, si ce n'est la plus célèbre : elle est en tout cas considérée comme l'une des représentations d'un visage féminin les plus célèbres au monde. Au xxie siècle, elle est devenue l'objet d'art le plus visité au monde, devant le diamant Hope, avec 20 000 visiteurs qui viennent l'admirer et la photographier quotidiennement.

8. Faites en sorte que le texte soit intégré dans un container adapté aux dimensions de l'image (cf. captures d'écrans en annexe) et puisse défiler verticalement (scroll), dans le cas où le volume de texte le nécessite. Superposez le texte à l'image.

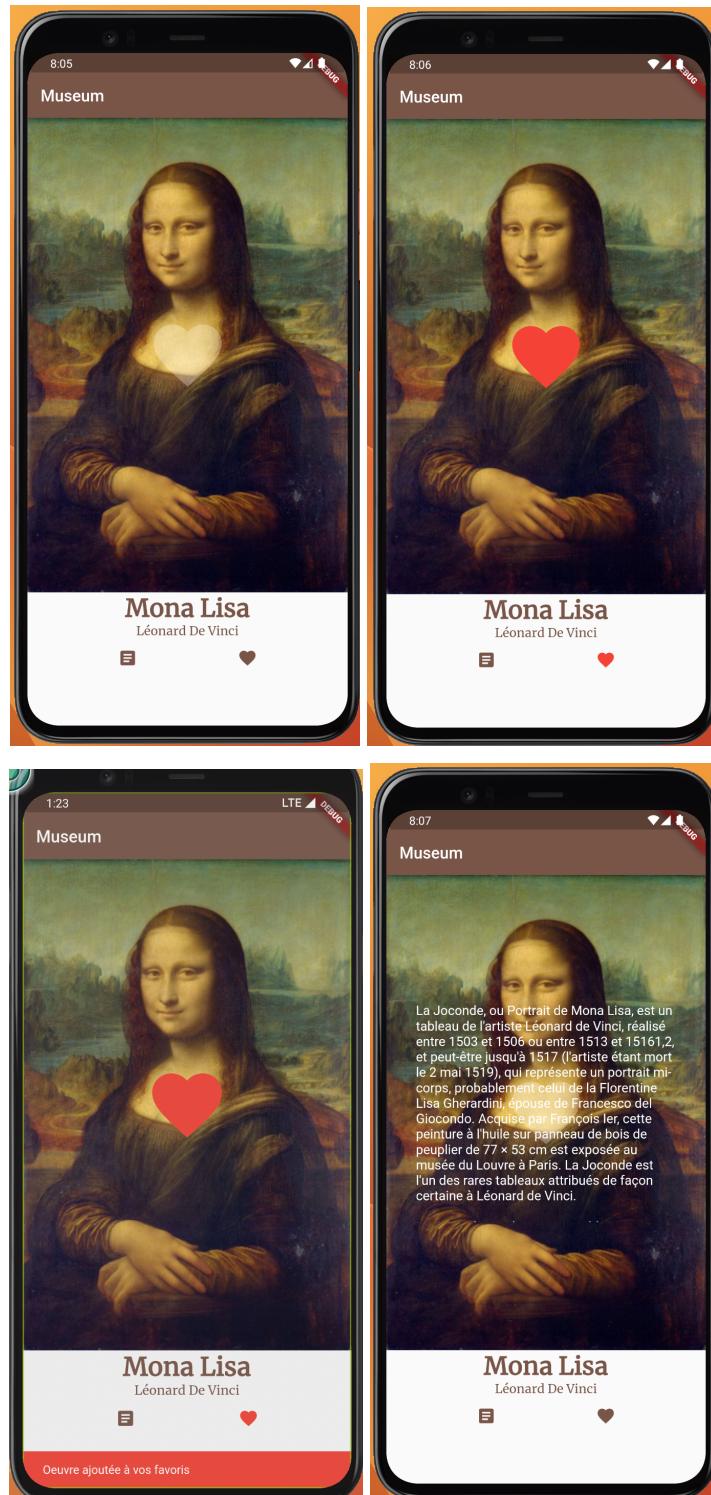
Etape 6 : Refactoring du code

- Procédez à une réorganisation du code de façon à disposer d'un seul fichier par entité du logiciel :
 - 1 fichier pour le main.
 - 1 fichier pour une classe de widget Stateless,
 - 1 fichier réunissant la classe de widget Stateful et sa classe State (à ne jamais séparer),
- Adaptez les imports en conséquence.

Structure de l'application



Captures d'écran



Crédits

- Texte et image : https://fr.wikipedia.org/wiki/La_Joconde
- TP : Alexandre Leroux.