



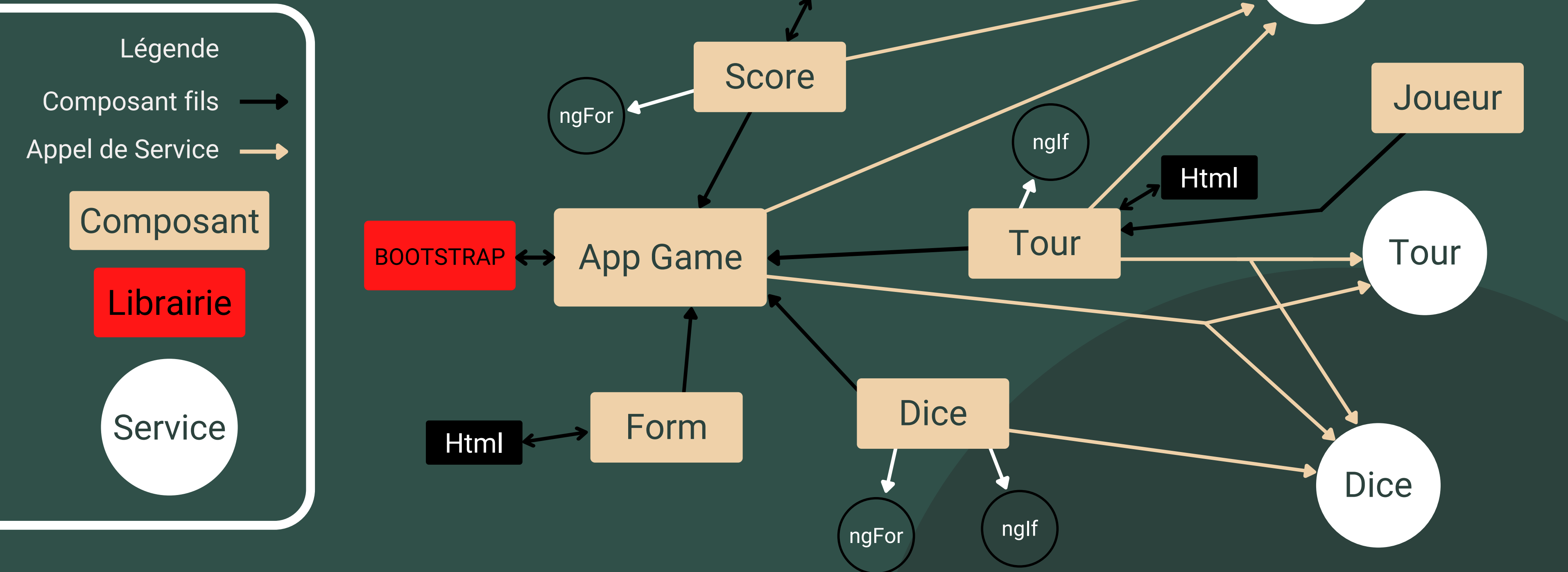
Jeu de Yam's

CARON SYLVAIN
CHANCHUS FLORIAN

Règles du jeu

Le brelan	Trois faces identiques	Trois fois la valeur de la face
Le carré	Quatre faces identiques	Quatre fois la valeur de la face
Le full	Un brelan et une paire	25 points
La petite suite	Une suite de quatre faces	30 points
La grande suite	Une suite de cinq faces	40 points
Le Yam's	Cinq faces identiques	50 points
La chance	Aucune des autres	Somme des cinq faces

Schéma



Composant fils

Appel de Service

Composant

Librairie

Service

Présentation du code

▼ src

▼ app

> dice

> form-player

> joueur

> tab-score

> tour

<> app.component.html

🔗 app.component.scss

TS app.component.sp...

TS app.component.ts

Composant

Fonctionnement

Dice

Regroupe le HTML et une Classe Dice pour la création des 5 dè

Form-Player

Formulaire pour les noms des joueurs, retourne les données vers App

Joueur

Une Classe Joueur pour la création des 2 joueurs

Tab-score

Tableau HTML qui affiche les scores retourné par le service des scores

Tour

Dispose d'un HTML et une Classe Tour pour création d'un objet Tour lors du début du jeu

App

Composent Parent qui regroupe les autres composants pour l'appel et l'affichage

Exemple d'un Composant :

✓ dice

❏ dice.component....

🔗 dice.component.s...

TS dice.component.s...

TS dice.component.ts

TS Dice.ts

← Html du Composant Dice pour l'affichage des dès

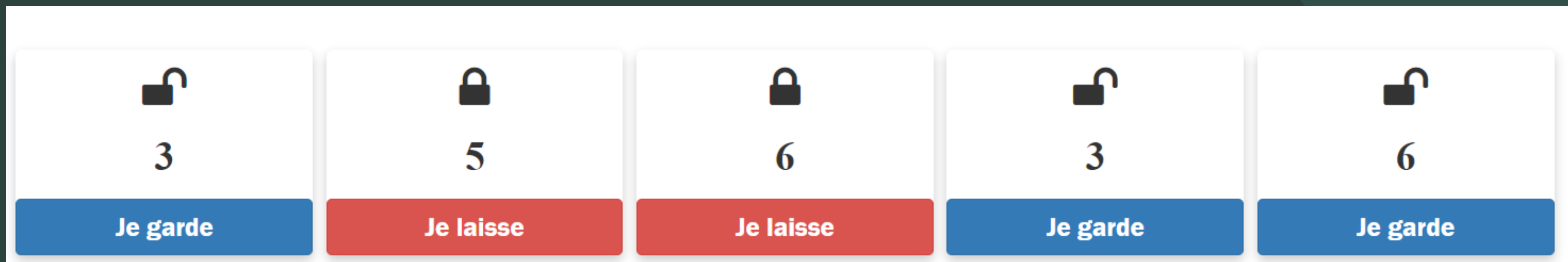
← Logique du Composant Dice qui fera appel aux services nécessaires

← Classe Dice pour la création d'un objet "Dès"

Utilisation des "ngFor" pour l'affiche des
dès
"ngIf" pour un affichage de texte ou de
boutons en fonction des dès "Bloquer"
/ "Débloquer"

```
<div class="row">
  <div class="col" style="margin-top: 25px;">
    <div *ngFor="let dice of allDice;" class="card" style="width: 19.1%;">
      <h5 *ngIf="dice.lock" class="card-header" style="text-align: center; font-size: 1.2em;">
        Bloquer
      <h5 *ngIf="!dice.lock" class="card-header" style="text-align: center; font-size: 1.2em;">
        Débloquer
      <div class="card-body">
        <p class="card-text" style="text-align:center;">
          <span *ngIf="dice.valeur == 0">?</span>
          <span *ngIf="dice.valeur != 0">{{ dice.valeur }}</span>
        </p>
        <button *ngIf="dice.lock" style="width: 100%; font-size: 20px;" class="btn btn-primary">
          Débloquer
        <button *ngIf="!dice.lock" style="width: 100%; font-size: 20px;" class="btn btn-primary">
          Bloquer
      </div>
    </div>
  </div>
</div>
```

Résultat :



Les classes

Présentation des classes disponible dans l'application Angular pour le Jeu de Yam's

- Dice :**
- Nom : String -> Stock le nom du dès
 - Valeur : Number -> Valeur du dès (par défaut 0)
 - Relance : Number -> Relance du dès (par défaut 3)
 - Lock : Boolean -> Bloquage du dès (par défaut False)
- Joueur :**
- Nom : String -> Stock le nom du joueur
 - Score : Number -> Stock le score du joueur
 - Tour : Number -> Stock le tour du joueur (max 10)
 - Relance : Number -> Stock la relance par joueur (max 3)
- Tour :**
- Tour : Number -> Tour N° (par défaut 1);
 - ActuelJoueur : Joueur -> Stock le joueur qui joue
 - PreviousJoueur : Joueur -> Stock le joueur du tour précédent

Les services

Les services vont effectués tous les calculs nécessaire sur les données puis les diffusés

TS `dice.service.spec.ts`

TS `dice.service.ts`

TS `score-calcul.service...`

TS `score-calcul.service...`

TS `tour.service.spec.ts`

TS `tour.service.ts`

Le service Dice permet de lancer les dès, bloqué/débloqué un dès et de voir la valeur d'un dès

Le service Score permet d'effectués tous les calculs sur la valeurs des dès et voir le résultat, il l'assigne ensuite au joueur et au tableau des scores

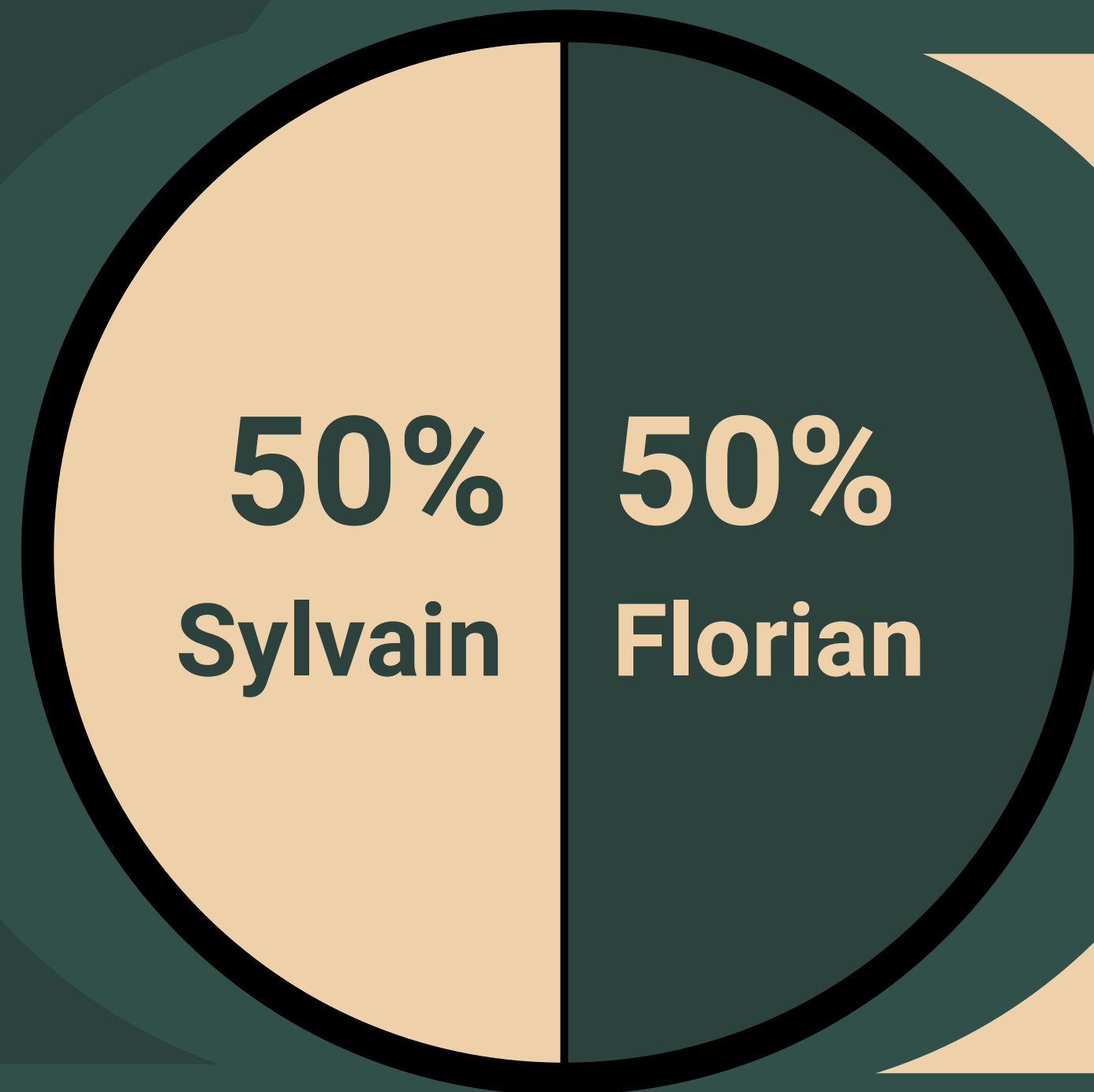
Le service Tour gère le changement de tour ainsi que le changement de joueur

Répartition du travail

PowerPont : 30%

Vidéo : 50%

Code : 70%



Code : 30%

Vidéo : 50%

PowerPont : 70%

Note du projet

Nous nous attribuons la note de 18/20.

Le projet est dans un premier temps très complet et répond à l'essentiel du cahier des charges d'un jeu de Yam's.

Cependant il serait possible de rajouter quelques fonctionnalités qui pourrait rendre le projet plus pertinent.

Comme la connexion à une base de données pour stocker les scores.



18/20