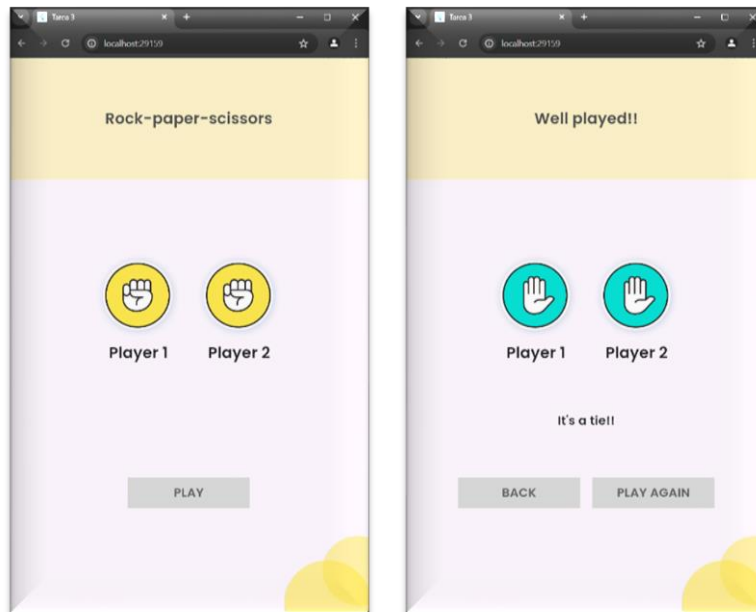


DESARROLLO MULTIPLATAFORMA Y WEB CON FLUTTER

(241107FP004)

Tarea 3: https://github.com/dcanoIESRafaelAlberti/Flutter_tarea3-main.git

He intentado refactorizarlo aún más generando un nuevo fichero (*utils_view.dart*) dónde he incluido las clases *HeaderView*, *BodyView*, *BottomView* y *Button*, dejándolas con el modificador de visibilidad por defecto (público), para que lo compartan las clases *GamePage* y *ResultPage*.



```
class GamePage extends StatelessWidget {
  const GamePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Column(
        mainAxisAlignment: MainAxisAlignment.spaceBetween,
        children: [
          const HeaderView("Rock-paper-scissors"),
          const SizedBox(height: 25,),
          const BodyView("rock", "rock"),
          const SizedBox(height: 96,),
          Button("PLAY", () => navigateToResultPage(context)),
          const BottomView()
        ],
      ), // Column
    ); // Scaffold
  }

  void navigateToResultPage(BuildContext context) {
    Navigator.of(context).push(MaterialPageRoute(
      builder: (_) {
        return ResultPage();
      },
    )); // MaterialPageRoute
  }
}
```

```

class ResultPage extends StatefulWidget {
  const ResultPage({super.key});

  @override
  State<ResultPage> createState() => _ResultPageState();
}

class _ResultPageState extends State<ResultPage> {
  final List<String> gamePlay = ["rock", "paper", "scissors"];
  int player1Play = 0;
  int player2Play = 0;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Column(
        mainAxisAlignment: MainAxisAlignment.spaceBetween,
        children: [
          const HeaderView("Well played!!"),
          const SizedBox(height: 25,),
          BodyView(gamePlay[player1Play], gamePlay[player2Play]),
          playerWinner(gamePlay[player1Play], gamePlay[player2Play]),
          Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Button("BACK", () => Navigator.of(context).pop()),
              const SizedBox(width: 20),
              Button("PLAY AGAIN", () => playAgain()),
            ]
          ), // Row
          const BottomView(),
        ], // Column
      ), // Scaffold
    );
  }

  void playAgain() {
    return setState(() {
      player1Play = Random().nextInt(3);
      player2Play = Random().nextInt(3);
    });
  }

  Widget playerWinner(String player1, String player2) {...}
}

```

En "util_view.dart" he incluido las clases refactorizadas para llamarlas desde "game_page.dart" y "result_page.dart".

En la clase "Button" he creado un atributo para pasarle una función lambda que ejecute la acción en el evento "onTap".

No sé si es la forma más correcta, he estado investigando y así es la forma que he encontrado de declararla y en principio funciona correctamente.

```

class HeaderView extends StatelessWidget {...}

class BodyView extends StatelessWidget {...}

class _GamePlay extends StatelessWidget {...}

class BottomView extends StatelessWidget {...}

class Button extends StatelessWidget {
  const Button(this.buttonText, this.onTapAction, {super.key});

  final String buttonText;
  final VoidCallback onTapAction;

  @override
  Widget build(BuildContext context) {
    return GestureDetector(
      onTap: onTapAction,
      child: Container(
        width: 200,
        height: 50,
        color: const Color(0xFFD9D9D9),
        alignment: Alignment.center,
        child: Text(
          buttonText,
          style: const TextStyle(
            fontFamily: "Poppins",
            fontSize: 20,
            fontWeight: FontWeight.w600,
            color: Color(0xFF5B5B5B)), // TextStyle
        ), // Text
      ), // Container
    ); // GestureDetector
  }
}

```

Otros ejemplos vistos en la sesión 4:

También he incluido los ejemplos realizados durante la sesión refactorizados. Es dónde estuve jugando para ver cómo podía refactorizar el botón pasándole la acción a realizar en una función lambda que he utilizado en la solución final de la tarea 3. El código está en la carpeta “ejemplos”.

