

Module Atelier - Architecture applicative (4/9)

Code Module	Durée	Titre Diplôme	Bloc de Compétences	Promotion	Auteur
ARCE842 - DEVE702	20h	EISI / RNCP 35584	Concevoir & Développer des solutions applicatives métiers	2023/2024	Julien COURAUD

8. Global Layout

Afin de partir sur une base graphique exploitable, voici un exemple de composant `Board` possédant une 'grid':

```
// app/components/board/board.component.js

import React from "react";
import { View, Text, StyleSheet } from 'react-native';

const OpponentInfos = () => {
  return (
    <View style={styles.opponentInfosContainer}>
      <Text>Opponent infos</Text>
    </View>
  );
};

const OpponentTimer = () => {
  return (
    <View style={styles.opponentTimerContainer}>
      <Text>Timer: </Text>
    </View>
  );
};

const OpponentScore = () => {
  return (
    <View style={styles.opponentScoreContainer}>
      <Text>Score: </Text>
    </View>
  );
};

const OpponentDeck = () => {
  return (
    <View style={styles.deckOpponentContainer}>
      <Text>OpponentDeck</Text>
    </View>
  );
};
```

```

const Grid = () => {

  return (
    <View style={styles.gridContainer}>
      <Text>Grid</Text>
    </View>
  );
};

const Choices = () => {

  return (
    <View style={styles.choicesContainer}>
      <Text>Choices</Text>
    </View>
  );
};

const PlayerDeck = () => {
  return (
    <View style={styles.deckPlayerContainer}>
      <Text>PlayerDeck </Text>
    </View>
  );
};

const PlayerInfos = () => {
  return (
    <View style={styles.playerInfosContainer}>
      <Text>Player Infos</Text>
    </View>
  );
};

const PlayerTimer = () => {

  return (
    <View style={styles.playerTimerContainer}>
      <Text>Timer: </Text>
    </View>
  );
};

const PlayerScore = () => {

  return (
    <View style={styles.playerScoreContainer}>
      <Text>PlayerScore</Text>
    </View>
  );
};

const Board = ({ gameState }) => {
  return (
    <View style={styles.container}>
      <View style={[styles.row, { height: '5%' }]}>
        <OpponentInfos />
        <View style={styles.opponentTimerScoreContainer}>
          <OpponentTimer />
          <OpponentScore />
        </View>
      </View>
      <View style={[styles.row, { height: '25%' }]}>
        <OpponentDeck />
      </View>
      <View style={[styles.row, { height: '40%' }]}>
        <Grid />
        <Choices />
      </View>
      <View style={[styles.row, { height: '25%' }]}>
        <PlayerDeck />
      </View>
    </View>
  );
};

```

```

        <View style={styles.row, { height: '5%' }}>
          <PlayerInfos />
          <View style={styles.playerTimerScoreContainer}>
            <PlayerTimer />
            <PlayerScore />
          </View>
        </View>
      </View>
    );
  };

```

```

const styles = StyleSheet.create({
  container: {
    flex: 1,
    flexDirection: 'column',
    justifyContent: 'space-between',
    alignItems: 'center',
    width: '100%',
    height: '100%',
  },
  row: {
    flexDirection: 'row',
    width: '100%',
    borderBottomWidth: 1,
    borderColor: 'black',
  },
  opponentInfosContainer: {
    flex: 7,
    justifyContent: 'center',
    alignItems: 'center',
    borderRightWidth: 1,
    borderColor: 'black',
    backgroundColor: "lightgrey"
  },
  opponentTimerScoreContainer: {
    flex: 3,
    flexDirection: 'column',
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: "lightgrey"
  },
  opponentTimerContainer: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },
  opponentScoreContainer: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },
  deckOpponentContainer: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
    borderBottomWidth: 1,
    borderColor: "black"
  },
  gridContainer: {
    flex: 7,
    justifyContent: 'center',
    alignItems: 'center',
    borderRightWidth: 1,
    borderColor: 'black',
  },
  choicesContainer: {
    flex: 3,
    justifyContent: 'center',
    alignItems: 'center',
  },
  deckPlayerContainer: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },

```

```

        borderBottomWidth: 1,
        borderColor: 'black',
    },
    playerInfosContainer: {
        flex: 7,
        justifyContent: 'center',
        alignItems: 'center',
        borderRightWidth: 1,
        borderColor: 'black',
        backgroundColor: "lightgrey"
    },
    playerTimerScoreContainer: {
        flex: 3,
        flexDirection: 'column',
        justifyContent: 'center',
        alignItems: 'center',
        backgroundColor: "lightgrey"
    },
    playerTimerContainer: {
        flex: 1,
        justifyContent: 'center',
        alignItems: 'center',
        backgroundColor: "lightgrey"
    },
    playerScoreContainer: {
        flex: 1,
        justifyContent: 'center',
        alignItems: 'center',
        backgroundColor: "lightgrey"
    },
    },
});

export default Board;

```

Dans la suite du module, chacun des composants implémentés au sein du Board devra être externalisé dans un composant à part (Dossiers dans /components/board) avec sa déclaration et sa logique graphique (StyleSheets containers).