

Un componente propio podría ser, por ejemplo, MagoDestacado.tsx incluyendo los imports necesarios, como el mismísimo react, el view, text, stylesheet, image...

Después, definir estos props:

```
type Props = {  
  nombre: string;  
  descripcion: string;  
  imagen: any;  
};
```

Para después, exportando la constante del MagoDestacadoCard usé el React.FC (FunctionComponent) para darle los atributos a la imagen, como la búsqueda de esta con el source, el View, y los estilos del title abreviado a nombre y lo mismo con desc, abreviado a descripción (se que no se diría abreviado como tal, pero fue lo único que se me vino a la cabeza).

Después es sencillo, simplemente añadí lo que quería poner de cada style:

```
const styles = StyleSheet.create({  
  card: {  
    flexDirection: 'row',  
    backgroundColor: '#fff',  
    borderColor: '#000',  
    borderWidth: 1,  
    borderRadius: 10,  
    marginHorizontal: 16,  
    marginBottom: 16,  
    overflow: 'hidden',  
  },  
  image: {  
    width: 100,  
    height: 100,  
    resizeMode: 'cover',  
  },  
  info: {  
    flex: 1,  
    padding: 10,  
    justifyContent: 'center',  
  },  
  title: {  
    fontWeight: 'bold',  
    fontSize: 16,  
    marginBottom: 4,  
  },  
  desc: {  
    fontSize: 13,  
    color: '#555',  
  },  
});
```

Aquí le estoy diciendo que todo lo que se ubique en el StyleSheet, que cree los estilos con el .create para card, image, info, title y desc. Y ya lo que tiene dentro cada estilo va acorde a la captura, que no está igual que el ejercicio porque cuando le pongo un background-image la app no cambia, estuvo 2 horas para intentar arreglarlo pero nada.