

## LAB 1

### MỤC TIÊU

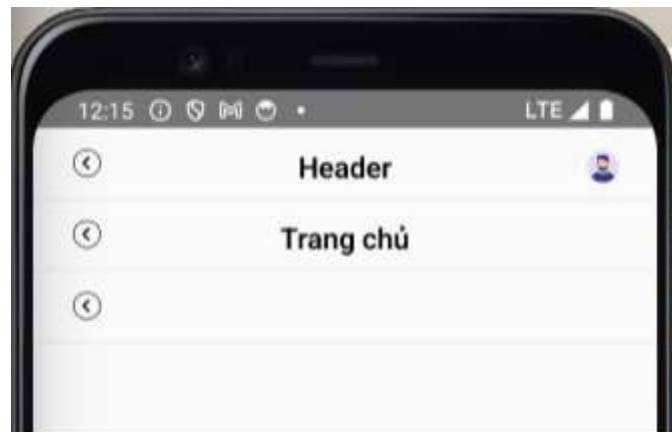
Kết thúc bài thực hành sinh viên có khả năng:

- ✓ Hiểu về khái niệm custom component.
- ✓ Thực hành, tạo nên các custom component hữu ích.

### NỘI DUNG

#### BÀI 1: TẠO HEADER CUSTOM COMPONENT

,



Yêu cầu:

- Tạo Header custom component.
- Hiện thị 3 cái Header có giao diện như trên

Hướng dẫn:

- Viết function renderLeft để hiển thị component bên trái.

```
const renderLeft = () => {
  return (
    leftComponent || (
      <View>
        {iconLeft ? (
          <Pressable hitSlop={15} onPress={onPressLeft || Navigator.
            goback}>
            <Image
              source={iconLeft} ...
              tint={iconLeftColor}
            />
          </Pressable>
        ) : (
          <View style={{width: leftIconSize, height: leftIconSize}} />
        )}
      </View>
    )
  );
};
```

- Viết function renderCenter để hiển thị component ở giữa.

```
const renderCenter = () => {
  return (
    centerComponent || (
      <View style={styles.containerCenter}>
        <Text style={styles.title} numberOfLines={numberOfLines}>
          {title}
        </Text>
      </View>
    )
  );
};
```

- Viết function renderRight để hiển thị component bên phải

```
const renderRight = () => {  
  return (  
    rightComponent || (  
      <View style={styles.containerRight}>  
        {iconRight ? (  
          <Pressable hitSlop={15} onPress={onPressRight}>  
            </Pressable>  
          ) : (  
            <View style={{width: rightIconSize, height:  
              rightIconSize}} />  
          )}  
        </View>  
      )  
    );  
  };  
};
```

- Gộp chung 3 thành phần để hiển thị cho Header custom component.

```
return (  
  <View style={styles.container}>  
    {renderLeft()}  
    {renderCenter()}  
    {renderRight()}  
  </View>  
);
```

- Gọi Header custom component để sử dụng.

```
<Header
  title="Header"
  iconRight={{ ...
  }}
  iconRightColor={null}
/>
<Header title="Trang chủ" />
<Header />
```

## BÀI 2: TẠO SECTION VIEW CUSTOM COMPONENT



Yêu cầu:

- Tạo Section view custom component có giao diện như hình ảnh.

Hướng dẫn:

- Chuẩn bị danh sách dữ liệu **eventInfo**, sau đó render ra các item section.

```
<ScrollView style={styles.container}>
  {eventInfo?.map?.(renderSection)}
</ScrollView>
```

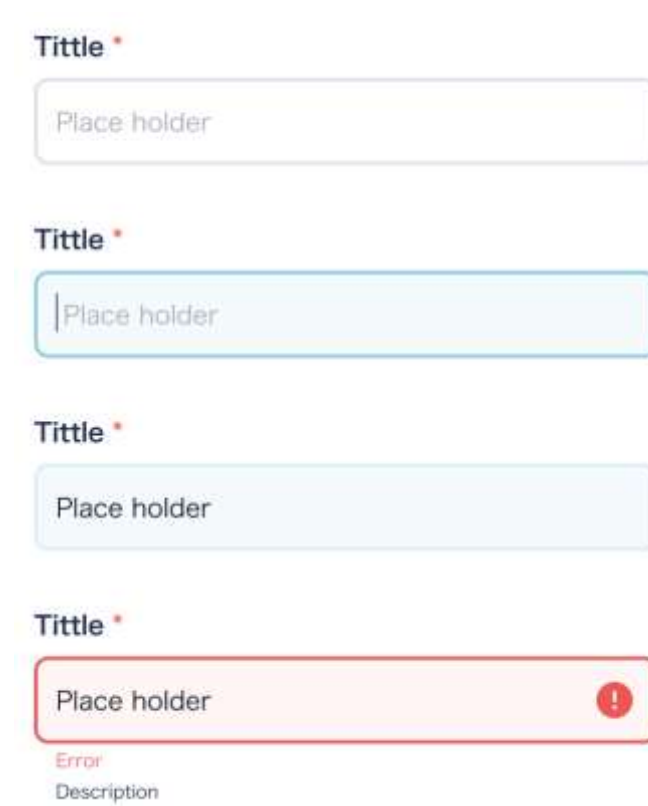
- Bên trong hàm **renderSection**, hiển thị tên tiêu đề và render các dữ liệu trong card

```
const renderSection = (data: SectionProps, index: number) => {
  const {title, events, titleStyle} = data;
  return (
    <View key={index.toString()} style={[styles.section]}>
      <Text style={[styles.titleSection, titleStyle]}>{title}</Text>
      <View style={[styles.sectionBody, styles.shadow]}>
        {events?.map(renderChild)}
      </View>
    </View>
  );
};
```

- Bên trong hàm **renderChild**, hiển thị tên tiêu đề và nội dung các dữ liệu trong section.

```
const renderChild = (data: EventProps, index: number) => {  
  const {  
  } = data;  
  
  return (  
    eventComponent || (  
      <View key={index.toString()} style={styles.containerChild}>  
        <Text style={[styles.titleChild, titleStyle]}>{title}</Text>  
        {contentComponent || (  
          <Text style={[styles.contentChild, contentStyle]}>{content}</Text>  
        )}  
      </View>  
    )  
  );  
};
```

### BÀI 3: TẠO TEXT INPUT CUSTOM COMPONENT



Yêu cầu:

- Tạo TextInput custom component hiển thị các trạng thái như hình ảnh.



Hướng dẫn:

- Tạo một **WrapInput** custom component để bọc lại **TextInput**. **WrapInput** chứa tiêu đề, lỗi và mô tả.

```
<WrapInput
  title={title}
  withColon={withColon}
  required={required}
  error={error}
  description={description}
  isFocus={isFocus}
  {...rest}>
  <View ...
</View>
```

- Lập trình các thành phần có trong TextInput như cấu trúc sau đây:

```
{...rest}>
<View style={[styles.container, style]}>
  <View style={styles.containerInput}>
    {renderLeft || renderLeftComponent()}
    <TextInput
      ref={inputRef} -
      autoCapitalize={autoCapitalize}
    />
    {showClear && !!value && ( ...
  )}
    {eyePassword && ( ...
  )}
    {!!error && <Image source={images.error} margin={{ right: 12 }} />}
    {renderRight || renderRightComponent()}
  </View>
</View>
</WrapInput>
```

#### BÀI 4: GV CHO THÊM

#### \*\*\* YÊU CẦU NỘP BÀI:

Sv nén file bao gồm các yêu cầu đã thực hiện trên, nộp lms đúng thời gian quy định của giảng viên. Không nộp bài coi như không có điểm.

--- Hết