


| | | | |
|---|-----------------------|--|--|
|  | | | |
| | REQUERIMIENTO TÉCNICO | | |

| 1.- IDENTIFICACIÓN DEL REQUERIMIENTO | | | |
|--------------------------------------|--|---------|-----|
| Proyecto | Facturacion.cl | | |
| Folio | Actividad 4 | Versión | 1.0 |
| Nombre | Crear una interfaz que permita visualizar los datos de un usuario y a la vez eliminar su cuenta luego de una confirmación. | | |
| Nº de entrega | 1 | | |

| 2.- DATOS DE ENTREGA | |
|----------------------|-----------------|
| Fecha Entrega | 23-10-2024 |
| Nombre Analista | Nicolás Hidalgo |
| Observación | |

Crear Archivo en Flutter

| | | | |
|----------------------|-----------------|----------------------|-------------|
| <i>Acción</i> | - Crear Archivo | <i>Nombre</i> | main_screen |
|----------------------|-----------------|----------------------|-------------|

Definición de parámetros a recibir o utilizar:

- BuildContext context

Definición de lógica a realizar

- **Navegación entre vistas:** Implementar la lógica para navegar entre "Creación Cuenta" y "Listado" con la ayuda de botones y/o Widgets.
- **Menú Hamburguesa (Drawer):** Se debe añadir un menú en la barra que permita navegar entre las distintas vistas.

IMAGEN 1

```

class _UserDetailsState extends State<UserDetails> {
  Widget build(BuildContext context) {
    ), // Text
    centerTitle: true,
    iconTheme: const IconThemeData(color: Colors.white),
    backgroundColor: const Color(0xFF1A90D9),
  ), // AppBar
  body: Padding(
    padding: const EdgeInsets.symmetric(horizontal: 35.0, vertical: 55),
    child: Column(
      children: [
        UserDetailData(
          label: "Nombre:",
          data: widget.user.name,
        ), // UserDetailData
        UserDetailData(
          label: "Correo Electronico:",
          data: widget.user.email,
        ), // UserDetailData
        UserDetailData(
          label: "Dirección:",
          data: widget.user.address,
        ), // UserDetailData
        UserDetailData(
          label: "Numero teléfono:",
          data: widget.user.phoneNumber,
        ), // UserDetailData
        UserDetailData(
          label: "Fecha de Nacimiento:",
          data: widget.user.birthDate,
        ), // UserDetailData
        UserDetailData(
          label: "Contraseña:",
          data: widget.user.password,
        ), // UserDetailData
        const SizedBox(
          height: 55,
        ), // SizedBox
        CustomButton(
          onPressed: () {
            _showAlert(context);
          },
          label: "Eliminar Cuenta",
          width: MediaQuery.of(context).size.width,
          height: 70,
          color: Colors.red[400]!,
          isEnabled: isLoading,
          isLoading: isLoading,
        ) // CustomButton
      ],
    ), // Column
  ), // Padding
); // Scaffold
}

```

IMAGEN 2

```

void _showAlert(BuildContext context) async {
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return AlertDialog(
        shape: const RoundedRectangleBorder(
          borderRadius: BorderRadius.all(Radius.circular(20.0)),
        ), // RoundedRectangleBorder
        title: const Center(
          child: Text(
            'Eliminar Cuenta',
            style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold),
          ), // Text // Center
        content: const Text(
          '¿Estas seguro de eliminar esta cuenta?',
          textAlign: TextAlign.justify,
          style: TextStyle(
            fontSize: 20,
          ), // TextStyle
        ), // Text
        actions: [
          ElevatedButton(
            onPressed: () {
              Navigator.pop(context);
            },
            style: ElevatedButton.styleFrom(
              backgroundColor: Colors.red[400]!,
              minimumSize: const Size(100, 35),
            ),
            child: const Text(
              "No",
              style: TextStyle(color: Colors.white),
            ), // Text
          ), // ElevatedButton
          ElevatedButton(
            onPressed: () {
              Navigator.pop(context);
              delete();
            },
            style: ElevatedButton.styleFrom(
              backgroundColor: const Color(0xFF1A90D9),
              minimumSize: const Size(100, 35),
            ),
            child: const Text(
              "Si",
              style: TextStyle(color: Colors.white),
            ), // Text
          ), // ElevatedButton
        ],
        actionsAlignment: MainAxisAlignment.center,
      ); // AlertDialog
    },
  );
}

```

Requerimiento #####

| |
|--|
| |
|--|

| Crear función para eliminar un usuario | | | |
|---|-----------------|---------------|------------------------|
| Acción | - Crear función | Nombre | deleteUser(int userId) |
| Definición de parámetros a recibir o utilizar: <ul style="list-style-type: none"> int userId | | | |
| Definición de lógica a realizar <ul style="list-style-type: none"> Eliminar usuario: Permite eliminar registros de la base de datos según la id del usuario. | | | |
| IMAGEN 1 | | | |
| <pre>Future<void> deleteUser(int userId) async { final db = await database; try { await db.delete('users', where: 'id = ?', whereArgs: [userId]); } catch (error) { log("\$error"); } }</pre> | | | |

| Crear función para eliminar un usuario | | | |
|---|-----------------|---------------|----------|
| Acción | - Crear función | Nombre | delete() |
| Definición de parámetros a recibir o utilizar: <ul style="list-style-type: none"> Database database | | | |
| Definición de lógica a realizar <ul style="list-style-type: none"> Eliminar cuenta: Al presionar el botón de eliminar cuenta y haber confirmado en el Dialog, se procede a eliminar al usuario de la base de datos llamando a la función mencionada anteriormente. | | | |
| IMAGEN 1 | | | |

```
void delete() async {  
  database.deleteUser(widget.user.id!);  
  
  if (mounted) {  
    Navigator.push(context, MaterialPageRoute(  
      builder: (context) {  
        return const UserListScreen();  
      },  
    )); // MaterialPageRoute  
    MessagesStatus.showStatusMessage(  
      context, "Cuenta eliminada con exito", false);  
  }  
}
```