UIP IA-2 (Content Creation)

Group-7

Kushal Panchal - 16010421067

Harsh Pandey - 16010421068

Keyur Patel - 16010421073

Problem statement

Background: Medinobel is a healthcare app that provides a platform for users to schedule appointments, search for medical professionals, and access healthcare-related information. To enhance the user experience and ensure accessibility for all users, we have designed a user-friendly and aesthetically pleasing components using Material Design principles.

Objective: Design and implement a user registration form for the Medinobel healthcare app using Dart Flutter(Material Design) components and principles. The form should include various input fields and interactive elements, such as radio buttons, checkboxes, date pickers, and text inputs, to improve usability and user engagement.

Requirements:

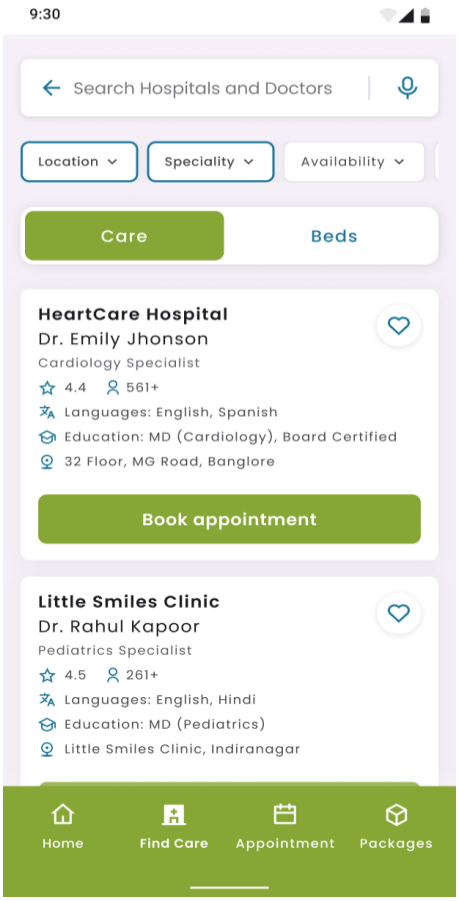
- **Material Design Components:** Utilize Material Design components like radio buttons, checkboxes, and date pickers to enhance the app usability and aesthetics. These components should seamlessly integrate with the overall app design.
- **Label Form Inputs:** Use label form inputs to provide a clear and concise description of each input field. Additionally, provide placeholder text to guide users in filling out the form.
- **Accessibility:** Ensure that the form fields are designed with accessibility in mind, making it usable by all users, including those with disabilities. This includes features like proper focus management and ARIA attributes for screen readers.

- **Validation:** Implement input validation for all form fields. Display error messages and indicators using Material UI components when user input is invalid. Validation should cover aspects such as required fields, proper format for dates, and selection of options.

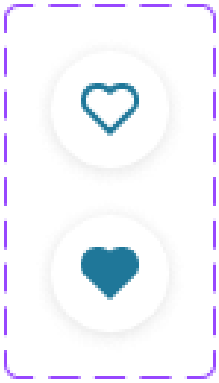
Interactive Examples and Code Snippets

1. **Code snippet for Material design components like radio buttons, checkboxes, and date pickers to improve usability**

   a) **Checkbox(used for marking favourite hospital during booking appointment)**



Searching for Hosptal to book an appointment



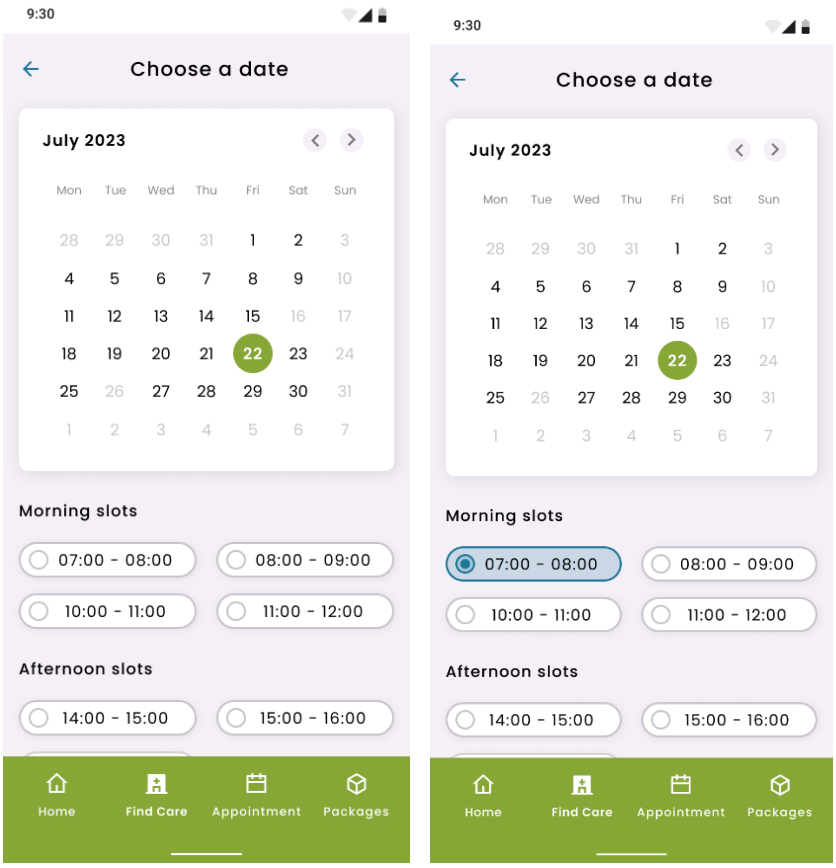
Checkbox Component(Heart icon)

Code Snippet

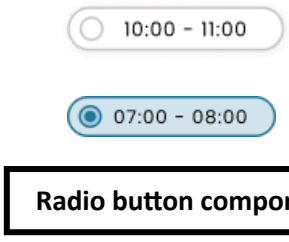
```
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text('Healthcare App'),
        ),
        body: FavoriteHospitalScreen(),
      ),
    );
  }
}

class FavoriteHospitalScreen extends StatefulWidget {
  @override
  _FavoriteHospitalScreenState createState() => _FavoriteHospitalScreenState();
}

class _FavoriteHospitalScreenState extends State<FavoriteHospitalScreen> {
  bool isFavorite = false;

  void toggleFavorite() {
    setState(() {
      isFavorite = !isFavorite;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          IconButton(
            icon: Icon(
              isFavorite ? Icons.favorite : Icons.favorite_border,
              color: isFavorite ? Colors.red : null,
              size: 48,
```

```
            ),
            onPressed: toggleFavorite,
          ),
          Text(
            isFavorite ? 'Favorited' : 'Not Favorited',
            style: TextStyle(fontSize: 20),
          ),
        ],
      ),
    );
  }
}
```

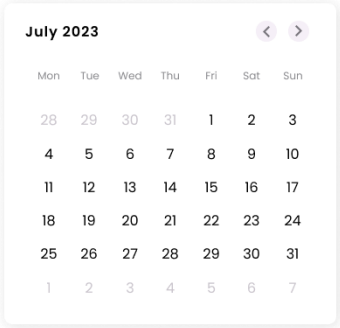
b) Radio button and date picker(used for marking the time slot of appointment and date picker is used for choosing the date of appointment)



Appointment Selection Widget



Radio button component



Date Picker component

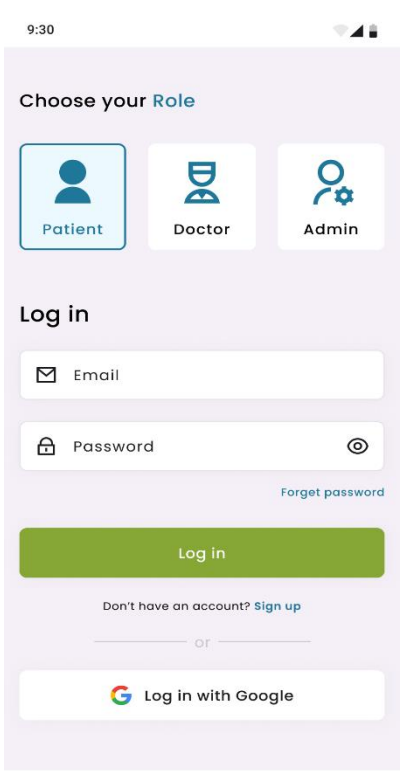
Code Snippet

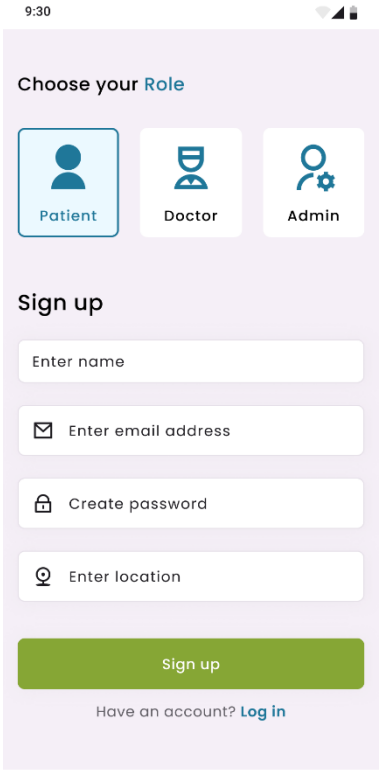
```dart
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text('Healthcare App'),
        ),
        body: AppointmentScreen(),
      ),
    );
  }
}

class AppointmentScreen extends StatefulWidget {
  @override
  _AppointmentScreenState createState() => _AppointmentScreenState();
}

class _AppointmentScreenState extends State<AppointmentScreen> {
  DateTime selectedDate = DateTime.now();
  String selectedTimeSlot = '';

  Future<void> _selectDate(BuildContext context) async {
    final DateTime picked = await showDatePicker(
      context: context,
      initialDate: selectedDate,
      firstDate: DateTime.now(),
      lastDate: DateTime(2030),
    );
    if (picked != null && picked != selectedDate)
      setState(() {
        selectedDate = picked;
      });
  }

  @override
  Widget build(BuildContext context) {
    return Center(
```

```
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          Text(
            'Choose Appointment Date:',
            style: TextStyle(fontSize: 18),
          ),
          SizedBox(height: 10),
          Text(
            "${selectedDate.toLocal()}".split(' ')[0],
            style: TextStyle(fontSize: 55, fontWeight: FontWeight.bold),
          ),
          SizedBox(
            height: 20.0,
          ),
          ElevatedButton(
            onPressed: () => _selectDate(context),
            child: Text(
              'Select date',
              style: TextStyle(color: Colors.white),
            ),
          ),
          SizedBox(
            height: 20.0,
          ),
          Text(
            'Choose Time Slot:',
            style: TextStyle(fontSize: 18),
          ),
          SizedBox(height: 10),
          RadioListTile<String>(
            title: Text('Morning Slot'),
            value: 'Morning Slot',
            groupValue: selectedTimeSlot,
            onChanged: (value) {
              setState(() {
                selectedTimeSlot = value;
              });
            },
          ),
          RadioListTile<String>(
            title: Text('Afternoon Slot'),
            value: 'Afternoon Slot',
            groupValue: selectedTimeSlot,
            onChanged: (value) {
              setState(() {
                selectedTimeSlot = value;
```

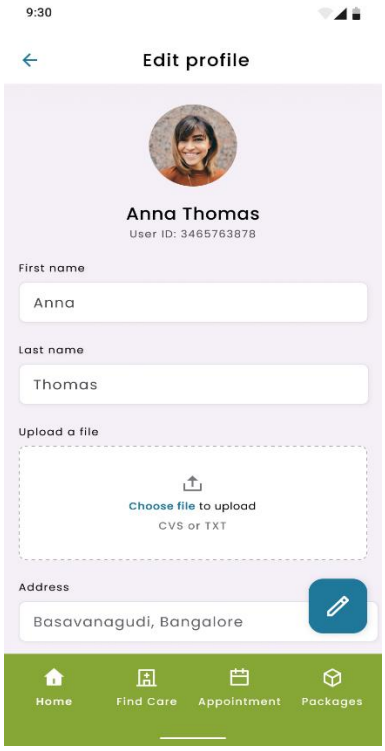
```
            });
          },
        ),
        RadioListTile<String>(
          title: Text('Evening Slot'),
          value: 'Evening Slot',
          groupValue: selectedTimeSlot,
          onChanged: (value) {
            setState(() {
              selectedTimeSlot = value;
            });
          },
        ),
        SizedBox(height: 20),
        Text(
          'Selected Time Slot: $selectedTimeSlot',
          style: TextStyle(fontSize: 20),
        ),
      ],
    ),
  );
 }
}
```

2. **Use label form inputs and provide placeholder text to guide users**





Login and signup widgets having placeholder text



Edit profile having label form inputs guiding the user

Code Snippet(for login and signup)

```dart
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text('Healthcare App'),
        ),
        body: LoginScreen(),
      ),
```

```
    );
  }
}

class LoginScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Center(
      child: Padding(
        padding: EdgeInsets.all(16),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            Text(
              'Login to Your Account',
              style: TextStyle(
                fontSize: 20,
                fontWeight: FontWeight.bold,
              ),
            ),
            SizedBox(height: 20),
            TextField(
              decoration: InputDecoration(
                labelText: 'Email',
                hintText: 'Enter your email',
                prefixIcon: Icon(Icons.email),
              ),
            ),
            SizedBox(height: 10),
            TextField(
              decoration: InputDecoration(
                labelText: 'Password',
                hintText: 'Enter your password',
                prefixIcon: Icon(Icons.lock),
              ),
              obscureText: true,
            ),
            SizedBox(height: 20),
            ElevatedButton(
              onPressed: () {
                // Perform login here
              },
              child: Text('Login'),
            ),
          ],
        ),
      ),
```

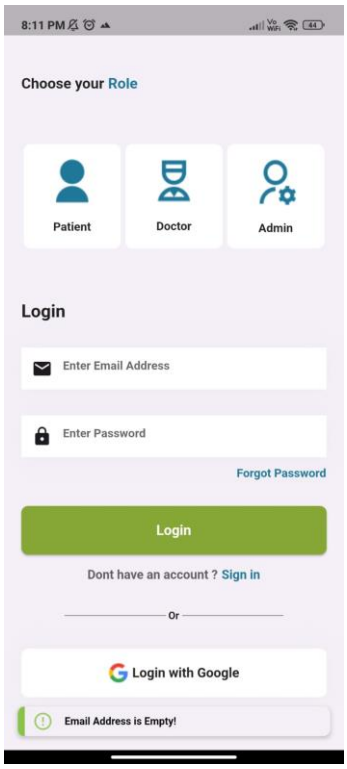
```
    );
  }
}
```

Code Snippet(for Edit profile)

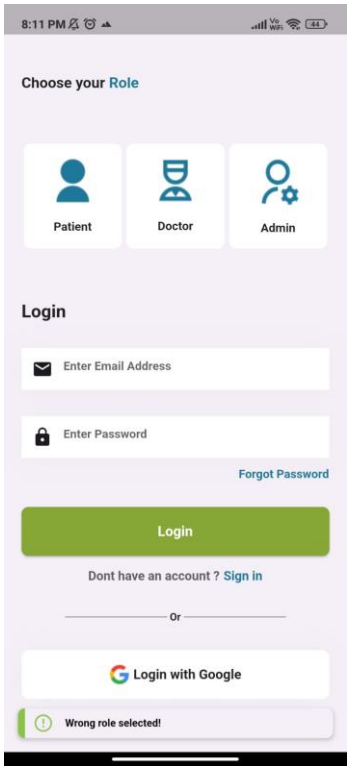
```
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text('Healthcare App'),
        ),
        body: EditProfileScreen(),
      ),
    );
  }
}
class EditProfileScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Center(
      child: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.center,
          children: <Widget>[
            CircleAvatar(
              radius: 60,
              backgroundImage: AssetImage('assets/profile_picture.png'), //
Replace with your image
            ),
            SizedBox(height: 20),
            Text(
              'Edit Profile',
              style: TextStyle(
                fontSize: 20,
                fontWeight: FontWeight.bold,
```

```
            ),
          ),
          SizedBox(height: 20),
          TextFormField(
            decoration: InputDecoration(
              labelText: 'First Name',
              hintText: 'Enter your first name',
              prefixIcon: Icon(Icons.person),
            ),
          ),
          SizedBox(height: 10),
          TextFormField(
            decoration: InputDecoration(
              labelText: 'Last Name',
              hintText: 'Enter your last name',
              prefixIcon: Icon(Icons.person),
            ),
          ),
          SizedBox(height: 10),
          ElevatedButton(
            onPressed: () {
              // Implement image upload logic
            },
            child: Text('Upload Profile Picture'),
          ),
          SizedBox(height: 10),
          TextFormField(
            decoration: InputDecoration(
              labelText: 'Address',
              hintText: 'Enter your address',
              prefixIcon: Icon(Icons.location_on),
            ),
            maxLines: 3,
          ),
          SizedBox(height: 20),
          ElevatedButton(
            onPressed: () {
              // Save profile changes
            },
            child: Text('Save Profile'),
          ),
        ],
      ),
    ),
  );
 }
}
```

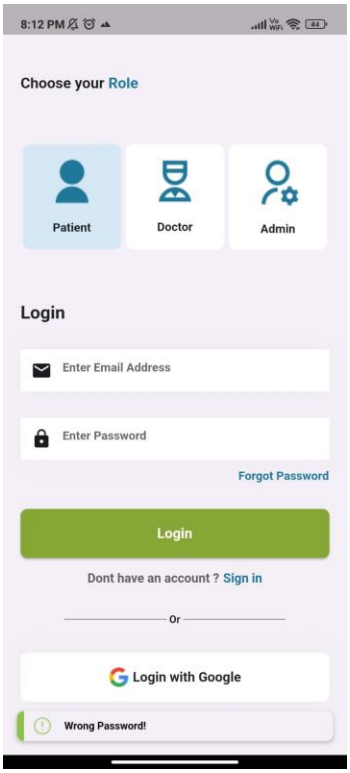
3. Validate user input and display errors using Material UI components(Login Form widget)



Validation for Email Address showing Email address is empty for patient role



Validation for wrong role selected.



Validation for Wrong password for patient role

Code Snippet(for validation of patient login widget)

```dart
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text('Healthcare App'),
        ),
        body: LoginScreen(),
      ),
    );
  }
```

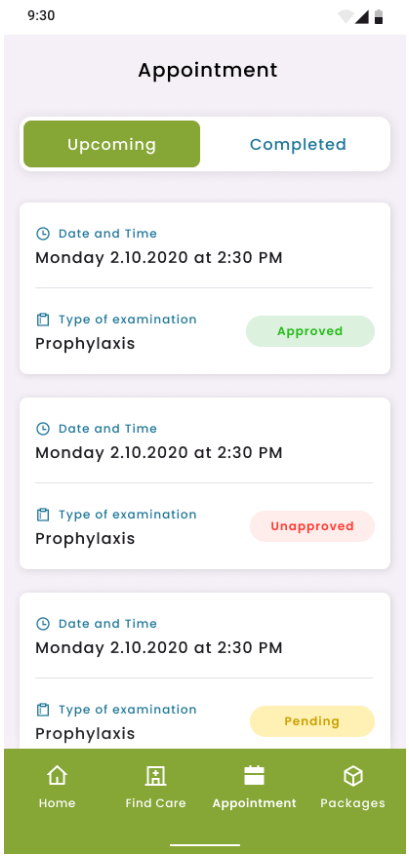
```
}

class LoginScreen extends StatefulWidget {
  @override
  _LoginScreenState createState() => _LoginScreenState();
}

class _LoginScreenState extends State<LoginScreen> {
  final _formKey = GlobalKey<FormState>();
  String email = '';
  String password = '';

  void _submitForm() {
    if (_formKey.currentState.validate()) {
      // Perform login or authentication here
      // You can use 'email' and 'password' variables
      // for your authentication logic
      // If authentication fails, display an error message.
    }
  }

  @override
  Widget build(BuildContext context) {
    return Center(
      child: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Form(
          key: _formKey,
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.center,
            children: <Widget>[
              Text(
                'Login to Your Account',
                style: TextStyle(
                  fontSize: 20,
                  fontWeight: FontWeight.bold,
                ),
              ),
              SizedBox(height: 20),
              TextFormField(
                decoration: InputDecoration(
                  labelText: 'Email',
                  hintText: 'Enter your email',
                  prefixIcon: Icon(Icons.email),
                ),
                validator: (value) {
                  if (value.isEmpty || !value.contains('@')) {
```

```
                    return 'Email Address is Empty!';
                  }
                  return null;
                },
                onSaved: (value) {
                  email = value;
                },
              ),
              SizedBox(height: 10),
              TextFormField(
                decoration: InputDecoration(
                  labelText: 'Password',
                  hintText: 'Enter your password',
                  prefixIcon: Icon(Icons.lock),
                ),
                obscureText: true,
                validator: (value) {
                  if (value.isEmpty) {
                    return 'Wrong Password!';
                  }
                  return null;
                },
                onSaved: (value) {
                  password = value;
                },
              ),
              SizedBox(height: 20),
              ElevatedButton(
                onPressed: _submitForm,
                child: Text('Login'),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

4) Use Material design components to ensure that form fields are accessible to all users

- **For Typography we have used poppins font style for our Medinobel app with size 12,16,20 & 24 for more readability .**
- **We have also given spacing and layouts in our healthcare app for better consistency ,more appealing and easy to understand layout.**
- **We have also used semantic colours in our apps using color intentionally and thoughtfully to convey information and meaning while ensuring that the app remains accessible to a broad user base.**

Semantic color Example



Here semantic colors are used in Patient flow panel for upcoming appointment schedule widget.

- **Green color represents appointment is approved.**
- **Red color represents appointment is unapproved.**
- **Yellow color represents appointment is pending.**

References

- https://docs.flutter.dev/ui/design/material
- https://medium.com/@kamal.lakhani56/semantics-flutter-7ec16ab6ce67#:~:text=Semantics%20is%20a%20widget%20that%20annotates%20the%20widget%20tree%20with,the%20meaning%20of%20the%20application.
- https://m2.material.io/develop/flutter
- https://devtechnosys.com/insights/develop-a-healthcare-app/