

MPL Lab 3

Mayur Jaiswal

D15B

24

Aim: To include icons, images and fonts in App.

Theory: Flutter allows developers to customize their apps by adding **icons**, **images**, and **custom fonts**. These visual elements enhance the user interface and make the app more appealing and intuitive.

- **Icons:** Flutter provides built-in icons via the Icons class. Custom icon packs (like FontAwesome) can also be added.
- **Images:** Images can be local (from assets) or loaded from the internet.
- **Fonts:** You can use Google Fonts or any custom TTF font by declaring them in pubspec.yaml.

Steps:

1. Add Icons:

2. Add Local Images:

Step A: Create an assets/images/ folder and add image files (e.g. logo.png).

Step B: Declare in pubspec.yaml:

Step C: Use in app:

3. Add Custom Fonts:

Step A: Create assets/fonts/ and add .ttf files.

Step B: Declare in pubspec.yaml:

Step C: Use in Text widget:

Code :

Login Page:

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

```
class LoginScreen extends StatefulWidget {
  @override
```

```

_LoginScreenState createState() => _LoginScreenState();
}

```

```

class _LoginScreenState extends State<LoginScreen> {
  final TextEditingController emailController = TextEditingController();
  final TextEditingController passwordController = TextEditingController();
  bool isLoading = false;

```

```

  void loginUser(BuildContext context) async {
    setState(() => isLoading = true);
    try {
      await Provider.of<AuthProvider>(context, listen: false)
        .login(emailController.text, passwordController.text);
      Navigator.pushReplacementNamed(context, '/home');
    } catch (e) {
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          content: Text("Login Failed: ${e.toString()}"),
          backgroundColor: Colors.redAccent,
        ),
      );
    }
    setState(() => isLoading = false);
  }

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Container(
      width: double.infinity,
      decoration: BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end: Alignment.bottomCenter,
          colors: [Colors.blueAccent, Colors.lightBlue.shade200],
        ),
      ),
    child: Center(
      child: Padding(
        padding: EdgeInsets.all(20.0),
        child: Card(
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(20),
          ),
          elevation: 8,
          color: Colors.white,
          child: Padding(
            padding: EdgeInsets.all(25.0),

```

```

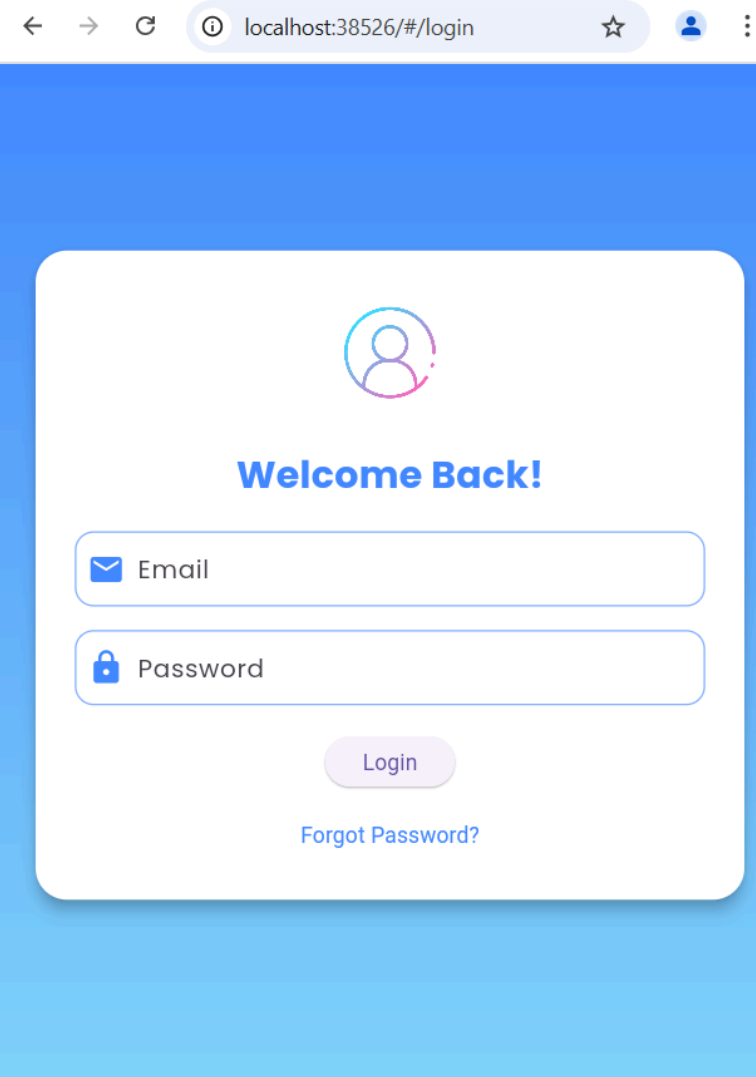
child: Column(
  mainAxisAlignment: MainAxisAlignment.min,
  children: [
    // Use a web image (URL)
    Image.network('https://cdn-icons-png.flaticon.com/512/5087/5087579.png', height: 80),
    SizedBox(height: 20),
    Text(
      "Welcome Back!",
      style: GoogleFonts.poppins(
        fontSize: 24,
        fontWeight: FontWeight.bold,
        color: Colors.blueAccent,
      ),
    ),
    SizedBox(height: 20),
    _buildTextField(
      controller: emailController,
      label: "Email",
      icon: Icons.email,
    ),
    SizedBox(height: 15),
    _buildTextField(
      controller: passwordController,
      label: "Password",
      isPassword: true,
      icon: Icons.lock,
    ),
    SizedBox(height: 20),
    isLoading
      ? CircularProgressIndicator()
      : CustomButton(
          text: "Login",
          onPressed: () => loginUser(context),
        ),
    SizedBox(height: 15),
    TextButton(
      onPressed: () {}, // Add forgot password logic
      child: Text(
        "Forgot Password?",
        style: TextStyle(color: Colors.blueAccent),
      ),
    ),
  ],
),
),
),
),
),
),
),
),
),
);

```

```
}
```


```
Widget _buildTextField(  
  {required TextEditingController controller,  
  required String label,  
  bool isPassword = false,  
  IconData? icon}) {  
  return TextField(  
    controller: controller,  
    obscureText: isPassword,  
    decoration: InputDecoration(  
      prefixIcon: icon != null ? Icon(icon, color: Colors.blueAccent) : null,  
      labelText: label,  
      labelStyle: GoogleFonts.poppins(fontSize: 16),  
      filled: true,  
      fillColor: Colors.white,  
      contentPadding: EdgeInsets.symmetric(horizontal: 20, vertical: 14),  
      border: OutlineInputBorder(  
        borderRadius: BorderRadius.circular(12),  
        borderSide: BorderSide(color: Colors.blueAccent),  
      ),  
      enabledBorder: OutlineInputBorder(  
        borderRadius: BorderRadius.circular(12),  
        borderSide: BorderSide(color: Colors.blueAccent.shade100),  
      ),  
      focusedBorder: OutlineInputBorder(  
        borderRadius: BorderRadius.circular(12),  
        borderSide: BorderSide(color: Colors.blueAccent, width: 2),  
      ),  
    ),  
  );  
}
```

Output:





A screenshot of a web browser displaying a login page. The browser's address bar shows the URL `localhost:38526/#/login`. The page has a blue gradient background. In the center, there is a white rounded rectangle containing a user icon, the text "Welcome Back!", an "Email" input field with an envelope icon, a "Password" input field with a lock icon, a "Login" button, and a "Forgot Password?" link.

localhost:38526/#/login



Welcome Back!

 Email

 Password

Login

[Forgot Password?](#)