MAD

In-Class Activity 13 (Part 2

The Interactive Flutter Signup Adventure

Due: 7:30 pm Today

**Class Activity Objective:**  
We will create an engaging and interactive signup experience in Flutter that goes beyond basic forms part 1 exercise. You'll build a multi-step signup process with animations, visual feedback, and a personalized welcome screen that makes users feel special!

**The Build Given:**

A fun, multi-screen signup journey featuring:

* **Screen 1:** Attractive signup form with animated transitions
* **Screen 2:** Personalized welcome screen with user's name
* **Visual rewards** for completing signup
* **Interactive feedback** throughout the process

**Step 1: Project Setup - Let's Make It Fun!**

*flutter create signup\_adventure*

*cd signup\_adventure*

**Step 2: Add Some Magic Dependencies**

Add these to your pubspec.yaml:

**yaml**

*dependencies:*

*flutter:*

*sdk: flutter*

*animated\_text\_kit: ^4.2.2*

*confetti: ^0.7.0*

**Step 3: The Adventure Begins - Main Dart File**

The main Dart file contains three screens:

1. **WelcomeScreen** - Animated landing page
2. **SignupScreen** - Interactive form with validation
3. **SuccessScreen** - Personalized celebration with confetti

**Your Mission - Level Up! (Challenges)- Let’s GOOOOOOOOO!!!!!!!!!!!!**

1. **Add Avatar Selection**
   * Let users choose from 3-5 fun avatars
   * Display selected avatar on success screen
2. **Password Strength Meter**
   * Create a visual indicator showing password strength
   * Change colors from red to green as password gets stronger
3. **Achievement Badges**
   * Award badges for: (use any creative logic here – images are fine)
     + "Strong Password Master" - Created strong password
     + "The Early Bird Special" (users who sign up before a certain time) – ie. signed up before 12 PM
     + "Profile Completer - Filled all fields
4. **Create: Implement "Adventure Progress Tracker"**

* **Requirements:**
* Create a visual progress bar showing signup completion (0% → 100%)
* Update progress in real-time as users complete each form field
* Add celebratory animations when reaching 25%, 50%, 75%, and 100% milestones
* Show encouraging messages at each milestone

**Implementation Hints:**

*// Starter code structure*

class ProgressTracker extends StatefulWidget {

*// Track completed fields*

*// Calculate percentage*

*// Animate progress changes*

}

*// Milestone messages:*

*// - 25%: "Great start! "*

*// - 50%: "Halfway there! "*

*// - 75%: "Almost done! "*

*// - 100%: "Ready for adventure! "*

1. **Bonus Challenge:**  (You must do any 2 of 4)

* Add sound effects or haptic feedback at each milestone!
* Bounce effect on valid input with checkmark icon
* Shake animation for invalid input with helpful tooltip
* Sequential field highlighting to guide users through form

**Example Animation Sequence:**

1. User types name → field glows green with ✅or anything cool
2. Valid email entered → field expands slightly with celebration emoji
3. Strong password → strength meter animates with sparkle effects

**Testing Checklist**

* Form shows creative error messages – Customize your messages
* Date picker opens calendar
* Password visibility toggle works
* Loading animation appears during submission
* Customize the Confetti celebration launches on success
* Personalized welcome message shows correct name

**Submission Instructions:**

1. **Build and Upload APK:**

flutter build apk --release

Upload the generated app-release.apk file

1. **GitHub Repository:**
   * Push all code to GitHub with regular commits
   * Include repository URL in submission
   * Show collaboration through commit history
2. **Required Submission:**
   * app-release.apk file
   * GitHub repository URL

**What are the Learning Outcomes:**

* Advanced form validation with user-friendly messages
* Navigation between multiple screens and separation of files into smaller chuncks
* Animation implementation
* State management with user interactions
* Date picking functionality
* Creating engaging user experiences

**You will implement the most exciting signup experience in Flutter! Your adventure awaits!**

*Pro Tip: Test your confetti celebration - it's the most satisfying part!*

*Note: Full code provided in separate document - split main.dart into multiple files for better organization*

// The Adventure Begins - Main Dart File

// Replace lib/main.dart with this exciting code:

import 'package:flutter/material.dart';

import 'package:animated\_text\_kit/animated\_text\_kit.dart';

import 'package:confetti/confetti.dart';

void main() {

  runApp(const SignupAdventureApp());

}

class SignupAdventureApp extends StatelessWidget {

  const SignupAdventureApp({super.key});

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      title: 'Signup Adventure ',

      theme: ThemeData(

        primarySwatch: Colors.deepPurple,

        fontFamily: 'Roboto',

      ),

      home: const WelcomeScreen(),

      debugShowCheckedModeBanner: false,

    );

  }

}

// Welcome Screen w/ Animated Text

class WelcomeScreen extends StatelessWidget {

  const WelcomeScreen({super.key});

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: Colors.deepPurple[50],

      body: Center(

        child: SingleChildScrollView(

          padding: const EdgeInsets.all(20.0),

          child: Column(

            mainAxisAlignment: MainAxisAlignment.center,

            children: [

              // Animated Emoji

              AnimatedContainer(

                duration: const Duration(seconds: 2),

                curve: Curves.bounceOut,

                width: 120,

                height: 120,

                decoration: BoxDecoration(

                  color: Colors.deepPurple,

                  shape: BoxShape.circle,

                  boxShadow: [

                    BoxShadow(

                      color: Colors.deepPurple.withOpacity(0.3),

                      blurRadius: 10,

                      spreadRadius: 2,

                    )

                  ],

                ),

                child: const Icon(

                  Icons.emoji\_emotions,

                  color: Colors.white,

                  size: 60,

                ),

              ),

              const SizedBox(height: 40),

              // Animated Title

              AnimatedTextKit(

                animatedTexts: [

                  TypewriterAnimatedText(

                    'Join The Adventure!',

                    textStyle: const TextStyle(

                      fontSize: 32,

                      fontWeight: FontWeight.bold,

                      color: Colors.deepPurple,

                    ),

                    speed: const Duration(milliseconds: 100),

                  ),

                ],

                totalRepeatCount: 1,

              ),

              const SizedBox(height: 20),

              const Text(

                'Create your account and start your journey',

                textAlign: TextAlign.center,

                style: TextStyle(fontSize: 16, color: Colors.grey),

              ),

              const SizedBox(height: 50),

              // Start Button

              ElevatedButton(

                onPressed: () {

                  Navigator.push(

                    context,

                    MaterialPageRoute(

                        builder: (context) => const SignupScreen()),

                  );

                },

                style: ElevatedButton.styleFrom(

                  backgroundColor: Colors.deepPurple,

                  padding:

                      const EdgeInsets.symmetric(horizontal: 40, vertical: 16),

                  shape: RoundedRectangleBorder(

                    borderRadius: BorderRadius.circular(30),

                  ),

                  elevation: 5,

                ),

                child: const Row(

                  mainAxisSize: MainAxisSize.min,

                  children: [

                    Text(

                      'Start Adventure',

                      style: TextStyle(fontSize: 18, color: Colors.white),

                    ),

                    SizedBox(width: 10),

                    Icon(Icons.arrow\_forward, color: Colors.white),

                  ],

                ),

              ),

            ],

          ),

        ),

      ),

    );

  }

}

// Signup Screen w/ Interactive Form

class SignupScreen extends StatefulWidget {

  const SignupScreen({super.key});

  @override

  State<SignupScreen> createState() => \_SignupScreenState();

}

class \_SignupScreenState extends State<SignupScreen> {

  final \_formKey = GlobalKey<FormState>();

  final TextEditingController \_nameController = TextEditingController();

  final TextEditingController \_emailController = TextEditingController();

  final TextEditingController \_passwordController = TextEditingController();

  final TextEditingController \_dobController = TextEditingController();

  bool \_isPasswordVisible = false;

  bool \_isLoading = false;

  @override

  void dispose() {

    \_nameController.dispose();

    \_emailController.dispose();

    \_passwordController.dispose();

    \_dobController.dispose();

    super.dispose();

  }

  // Date Picker Function

  Future<void> \_selectDate() async {

    DateTime? picked = await showDatePicker(

      context: context,

      initialDate: DateTime.now(),

      firstDate: DateTime(1900),

      lastDate: DateTime.now(),

    );

    if (picked != null) {

      setState(() {

        \_dobController.text = "${picked.day}/${picked.month}/${picked.year}";

      });

    }

  }

  void \_submitForm() {

    if (\_formKey.currentState!.validate()) {

      setState(() {

        \_isLoading = true;

      });

      // Simulate API call

      Future.delayed(const Duration(seconds: 2), () {

        if (!mounted) return; // Check if the widget is still in the tree

        setState(() {

          \_isLoading = false;

        });

        Navigator.pushReplacement(

          context,

          MaterialPageRoute(

            builder: (context) => SuccessScreen(userName: \_nameController.text),

          ),

        );

      });

    }

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: const Text('Create Your Account 🎉'),

        backgroundColor: Colors.deepPurple,

        elevation: 0,

      ),

      body: Padding(

        padding: const EdgeInsets.all(24.0),

        child: SingleChildScrollView(

          child: Form(

            key: \_formKey,

            child: Column(

              children: [

                // Animated Form Header

                AnimatedContainer(

                  duration: const Duration(milliseconds: 500),

                  curve: Curves.easeInOut,

                  padding: const EdgeInsets.all(16),

                  decoration: BoxDecoration(

                    color: Colors.deepPurple[100],

                    borderRadius: BorderRadius.circular(20),

                  ),

                  child: Row(

                    children: [

                      Icon(Icons.tips\_and\_updates,

                          color: Colors.deepPurple[800]),

                      const SizedBox(width: 10),

                      Expanded(

                        child: Text(

                          'Complete your adventure profile!',

                          style: TextStyle(

                            color: Colors.deepPurple[800],

                            fontWeight: FontWeight.bold,

                          ),

                        ),

                      ),

                    ],

                  ),

                ),

                const SizedBox(height: 30),

                // Name Field

                \_buildTextField(

                  controller: \_nameController,

                  label: 'Adventure Name',

                  icon: Icons.person,

                  validator: (value) {

                    if (value == null || value.isEmpty) {

                      return 'What should we call you on this adventure?';

                    }

                    return null;

                  },

                ),

                const SizedBox(height: 20),

                // Email Field

                \_buildTextField(

                  controller: \_emailController,

                  label: 'Email Address',

                  icon: Icons.email,

                  validator: (value) {

                    if (value == null || value.isEmpty) {

                      return 'We need your email for adventure updates!';

                    }

                    if (!value.contains('@') || !value.contains('.')) {

                      return 'Oops! That doesn\'t look like a valid email';

                    }

                    return null;

                  },

                ),

                const SizedBox(height: 20),

                // DOB w/Calendar

                TextFormField(

                  controller: \_dobController,

                  readOnly: true,

                  onTap: \_selectDate,

                  decoration: InputDecoration(

                    labelText: 'Date of Birth',

                    prefixIcon:

                        const Icon(Icons.calendar\_today, color: Colors.deepPurple),

                    border: OutlineInputBorder(

                      borderRadius: BorderRadius.circular(12),

                    ),

                    filled: true,

                    fillColor: Colors.grey[50],

                    suffixIcon: IconButton(

                      icon: const Icon(Icons.date\_range),

                      onPressed: \_selectDate,

                    ),

                  ),

                  validator: (value) {

                    if (value == null || value.isEmpty) {

                      return 'When did your adventure begin?';

                    }

                    return null;

                  },

                ),

                const SizedBox(height: 20),

                // Pswd Field w/ Toggle

                TextFormField(

                  controller: \_passwordController,

                  obscureText: !\_isPasswordVisible,

                  decoration: InputDecoration(

                    labelText: 'Secret Password',

                    prefixIcon:

                        const Icon(Icons.lock, color: Colors.deepPurple),

                    border: OutlineInputBorder(

                      borderRadius: BorderRadius.circular(12),

                    ),

                    filled: true,

                    fillColor: Colors.grey[50],

                    suffixIcon: IconButton(

                      icon: Icon(

                        \_isPasswordVisible

                            ? Icons.visibility

                            : Icons.visibility\_off,

                        color: Colors.deepPurple,

                      ),

                      onPressed: () {

                        setState(() {

                          \_isPasswordVisible = !\_isPasswordVisible;

                        });

                      },

                    ),

                  ),

                  validator: (value) {

                    if (value == null || value.isEmpty) {

                      return 'Every adventurer needs a secret password!';

                    }

                    if (value.length < 6) {

                      return 'Make it stronger! At least 6 characters';

                    }

                    return null;

                  },

                ),

                const SizedBox(height: 30),

                // Submit Button w/ Loading Animation

                AnimatedContainer(

                  duration: const Duration(milliseconds: 300),

                  width: \_isLoading ? 60 : double.infinity,

                  height: 60,

                  child: \_isLoading

                      ? const Center(

                          child: CircularProgressIndicator(

                            valueColor: AlwaysStoppedAnimation<Color>(

                                Colors.deepPurple),

                          ),

                        )

                      : ElevatedButton(

                          onPressed: \_submitForm,

                          style: ElevatedButton.styleFrom(

                            backgroundColor: Colors.deepPurple,

                            shape: RoundedRectangleBorder(

                              borderRadius: BorderRadius.circular(15),

                            ),

                            padding: const EdgeInsets.symmetric(vertical: 16),

                            elevation: 5,

                          ),

                          child: const Row(

                            mainAxisAlignment: MainAxisAlignment.center,

                            children: [

                              Text(

                                'Start My Adventure',

                                style: TextStyle(

                                    fontSize: 18, color: Colors.white),

                              ),

                              SizedBox(width: 10),

                              Icon(Icons.rocket\_launch, color: Colors.white),

                            ],

                          ),

                        ),

                ),

              ],

            ),

          ),

        ),

      ),

    );

  }

  Widget \_buildTextField({

    required TextEditingController controller,

    required String label,

    required IconData icon,

    required String? Function(String?) validator,

  }) {

    return TextFormField(

      controller: controller,

      decoration: InputDecoration(

        labelText: label,

        prefixIcon: Icon(icon, color: Colors.deepPurple),

        border: OutlineInputBorder(

          borderRadius: BorderRadius.circular(12),

        ),

        filled: true,

        fillColor: Colors.grey[50],

      ),

      validator: validator,

    );

  }

}

// Tha Big Confetti Celebration

class SuccessScreen extends StatefulWidget {

  final String userName;

  const SuccessScreen({super.key, required this.userName});

  @override

  State<SuccessScreen> createState() => \_SuccessScreenState();

}

class \_SuccessScreenState extends State<SuccessScreen> {

  late final ConfettiController \_confettiController;

  @override

  void initState() {

    super.initState();

    \_confettiController =

        ConfettiController(duration: const Duration(seconds: 10));

    \_confettiController.play();

  }

  @override

  void dispose() {

    \_confettiController.dispose();

    super.dispose();

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: Colors.deepPurple[50],

      body: Stack(

        children: [

          // Confetti Aniiiii

          Align(

            alignment: Alignment.topCenter,

            child: ConfettiWidget(

              confettiController: \_confettiController,

              blastDirectionality: BlastDirectionality.explosive,

              shouldLoop: false,

              colors: const [

                Colors.deepPurple,

                Colors.purple,

                Colors.blue,

                Colors.green,

                Colors.orange,

              ],

            ),

          ),

          // Tha Content

          Center(

            child: SingleChildScrollView(

              padding: const EdgeInsets.all(20.0),

              child: Column(

                mainAxisAlignment: MainAxisAlignment.center,

                children: [

                  // Celebration it is  Icon

                  AnimatedContainer(

                    duration: const Duration(seconds: 1),

                    curve: Curves.elasticOut,

                    width: 150,

                    height: 150,

                    decoration: BoxDecoration(

                      color: Colors.deepPurple,

                      shape: BoxShape.circle,

                      boxShadow: [

                        BoxShadow(

                          color: Colors.deepPurple.withOpacity(0.5),

                          blurRadius: 20,

                          spreadRadius: 5,

                        ),

                      ],

                    ),

                    child: const Icon(

                      Icons.celebration,

                      color: Colors.white,

                      size: 80,

                    ),

                  ),

                  const SizedBox(height: 40),

                  // Personalized Welcome Message

                  AnimatedTextKit(

                    animatedTexts: [

                      TypewriterAnimatedText(

                        'Welcome, ${widget.userName}! 🎉',

                        textAlign: TextAlign.center,

                        textStyle: const TextStyle(

                          fontSize: 28,

                          fontWeight: FontWeight.bold,

                          color: Colors.deepPurple,

                        ),

                        speed: const Duration(milliseconds: 100),

                      ),

                    ],

                    totalRepeatCount: 1,

                  ),

                  const SizedBox(height: 20),

                  const Text(

                    'Your adventure begins now!',

                    style: TextStyle(fontSize: 18, color: Colors.grey),

                  ),

                  const SizedBox(height: 50),

                  // Daaa... Continue Button

                  ElevatedButton(

                    onPressed: () {

                      \_confettiController.play();

                    },

                    style: ElevatedButton.styleFrom(

                      backgroundColor: Colors.deepPurple,

                      padding: const EdgeInsets.symmetric(

                          horizontal: 30, vertical: 15),

                      shape: RoundedRectangleBorder(

                        borderRadius: BorderRadius.circular(25),

                      ),

                    ),

                    child: const Text(

                      'More Celebration!',

                      style: TextStyle(fontSize: 16, color: Colors.white),

                    ),

                  ),

                ],

              ),

            ),

          ),

        ],

      ),

    );

  }

}

***Note: Full code provided in separate document - split main.dart into multiple files for better organization***

**See the Helper Document**