Login page validation

Exercise #3

Flutter Developer Bootcamp

# **Purpose**

This exercise demonstrates how to implement input validation and handling user input in a Flutter application, specifically in the context of a login screen.

**Problem**

In the provided Exercise You need to create as follows:

In the given exercise Creates a login screen with input validation. It consists of two text fields for username and password, with validation ensuring alphanumeric characters for the username and a minimum length of 8 characters for the password. Upon submission, valid inputs are printed to the console. The design includes an app bar titled "Login" and buttons for submission and clearing inputs. Error feedback is displayed below each field.

**How to Solve**

1. Checkout the exercise from Git Repo:

git clone -b <user-branch> <repo-URL>

2. Open the root folder inside VS Code

3. To build the app click the run option in the main method{}

4. Create Two TextField widgets are included for entering the username and password. They are decorated with InputDecoration to provide labels, hints, and error feedback based on validation status.

5. Create validateUsername and \_validatePassword Methods: These methods perform validation on the username and password inputs using regular expressions.

6. Create Two ElevatedButton widgets are included for submitting the form and clearing the inputs

7. Create Submit Method: \_submit validates inputs and prints them if valid.

8. Finally create Clear Method: \_clear resets inputs and validation status.

9. Go To File: <specific-file-with-validation-method> à <method-name>, implement your logic.

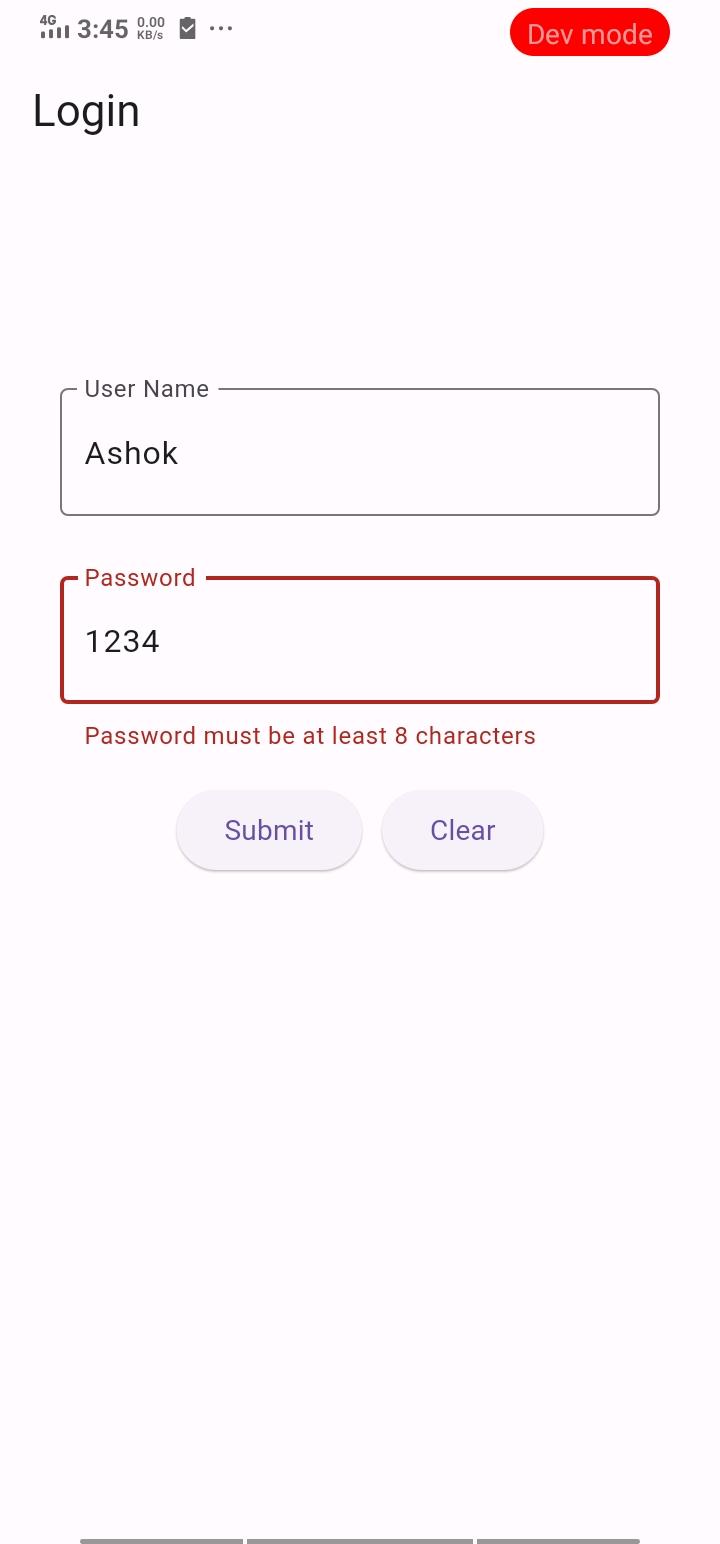
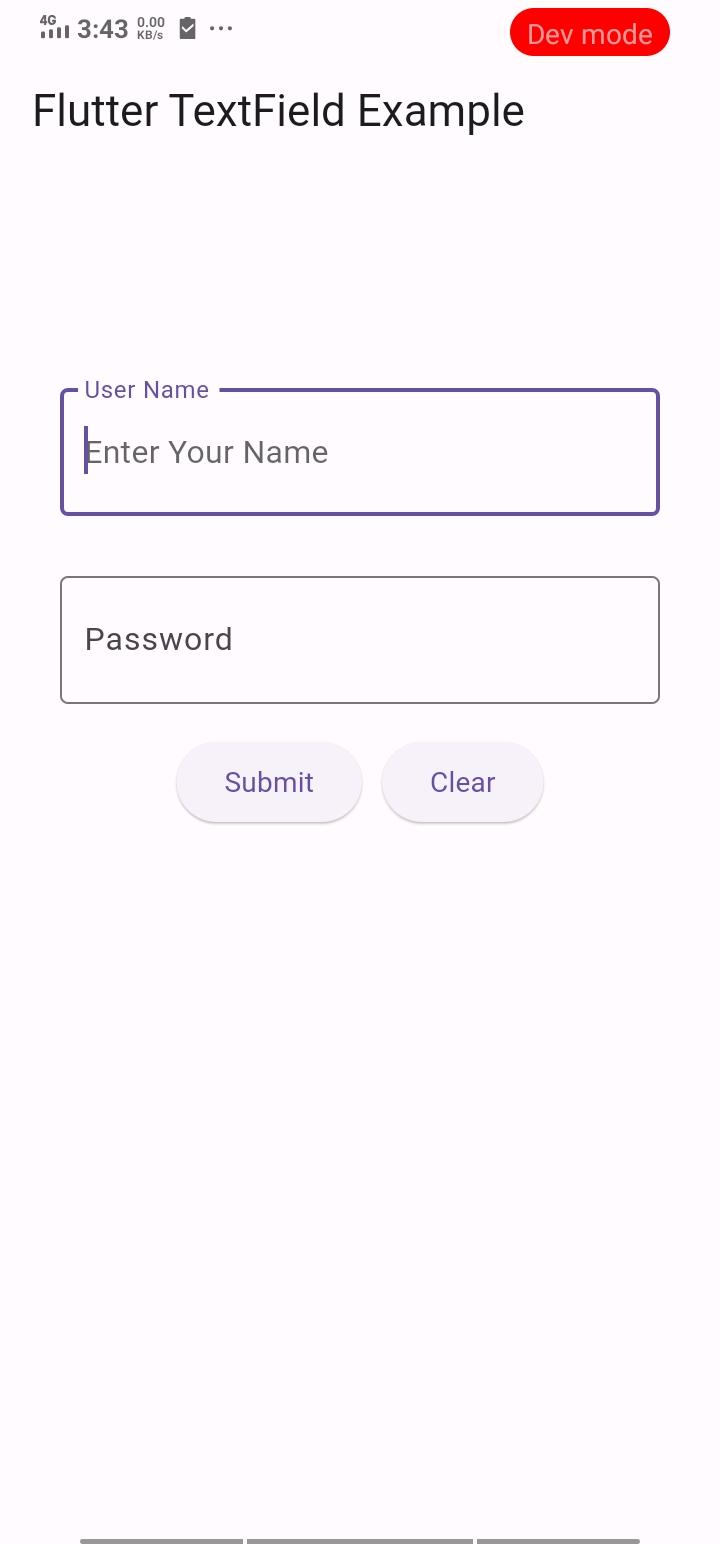
**You will Achieve**

When you complete this Exercise you will learn the following:

* Purpose:
* Provides a user interface for users to enter their username and password.
* Validates the input fields to ensure they meet certain criteria.
* Allows users to submit their credentials or clear the input fields.
* Included Widgets:
* **Scaffold**: Provides the overall structure for the screen, including app bar and body.
* **AppBar**: Displays the title "Login" at the top of the screen.
* **SingleChildScrollView**: Enables scrolling when the content exceeds the screen size.
* **Padding**: Adds padding around the content to provide spacing.
* **Column:** Organizes the child widgets vertically.
* **TextField**: Allows users to input text for username and password.
* **InputDecoration**: Customizes the appearance of the text fields, including border color and error text.
* **ElevatedButton**: Provides buttons for submitting and clearing input fields.
* **SizedBox**: Adds vertical spacing between widgets.
* **Row**: Arranges the buttons horizontally.

# **Screenshots**

## **Expected output( Login page validation)**



# **How to submit your Exercise**

Push your project back to the same git branch using command:

<command name>

# **Happy Coding!**