# EXPERIMENT 5

AIM: To implement navigation, routing, and gestures in a Flutter-based Expense Tracker app.

## THEORY:

In Flutter, building interactive and multi-screen applications involves the use of navigation, routing, and gesture handling. These features are essential for creating user-friendly, responsive, and intuitive mobile applications.  
  
Navigation refers to the ability to move between different screens in an app. Flutter uses a stack-based system, where new pages are pushed onto the stack and removed when the user navigates back.  
  
Routing defines how screens are organized and navigated between. There are two common types of routing in Flutter:  
- Basic Routing using MaterialPageRoute  
- Named Routing for structured navigation  
  
Gestures are user interactions such as taps, swipes, or long-presses. Flutter provides widgets like GestureDetector to detect and respond to these inputs.  
  
In the Expense Tracker app, navigation allows the user to log in and move to the home screen, view and add expenses, while gestures enable interactive elements like tapping an expense to view details.

## CODE SNIPPETS:

### Navigation: Login to Home

Navigator.pushReplacement(  
 context,  
 MaterialPageRoute(builder: (context) => HomeScreen()),  
);

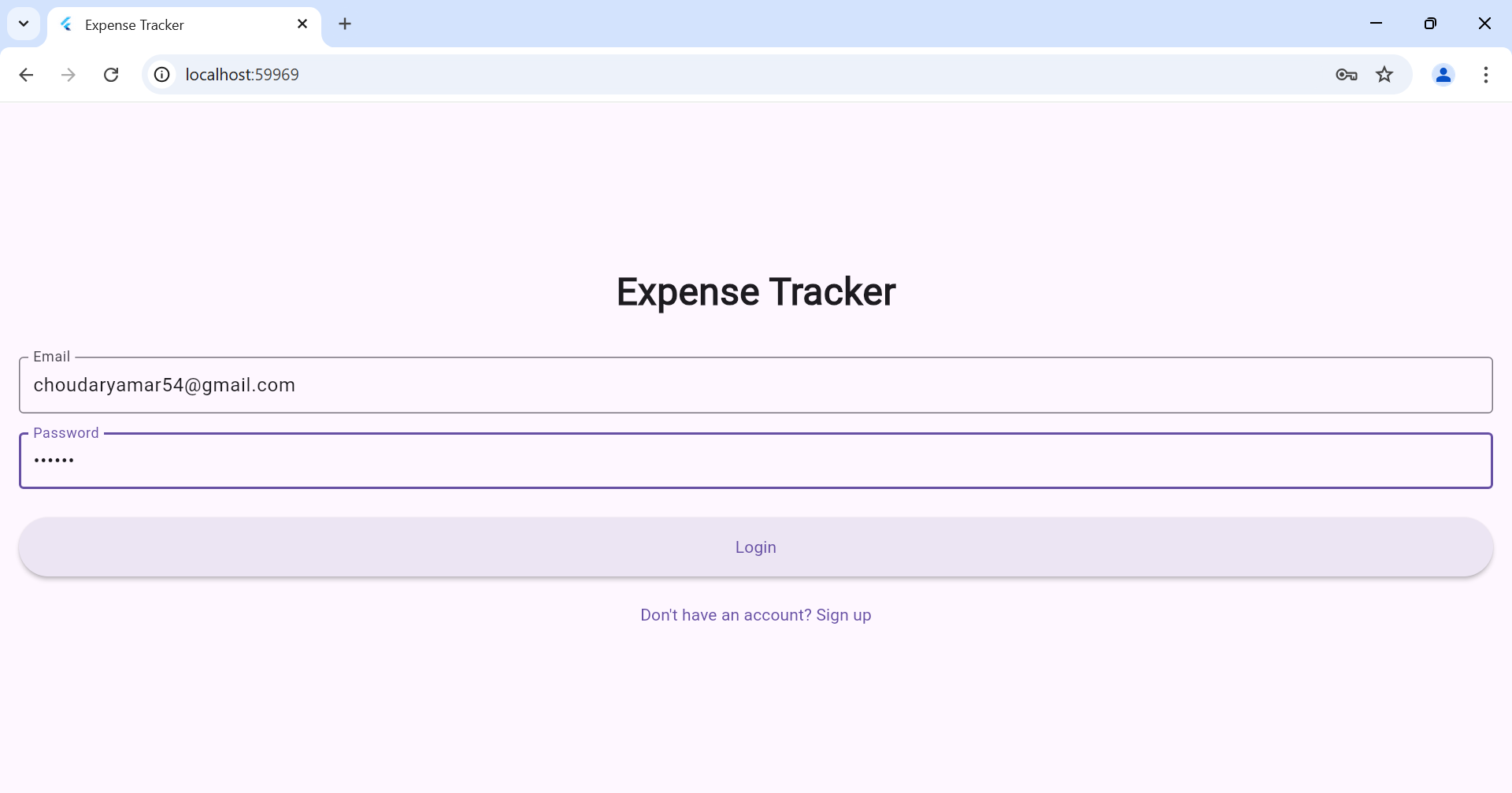
### Routing: Home to Add Expense

FloatingActionButton(  
 onPressed: () {  
 Navigator.push(  
 context,  
 MaterialPageRoute(builder: (context) => AddExpenseScreen()),  
 );  
 },  
 child: Icon(Icons.add),  
);

### Gesture: Tap to Show Details

GestureDetector(  
 onTap: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 title: Text("Expense Details"),  
 content: Text("Amount: ₹500  
Category: Food"),  
 ),  
 );  
 },  
 child: ExpenseCard(),  
);

## OUTPUT:



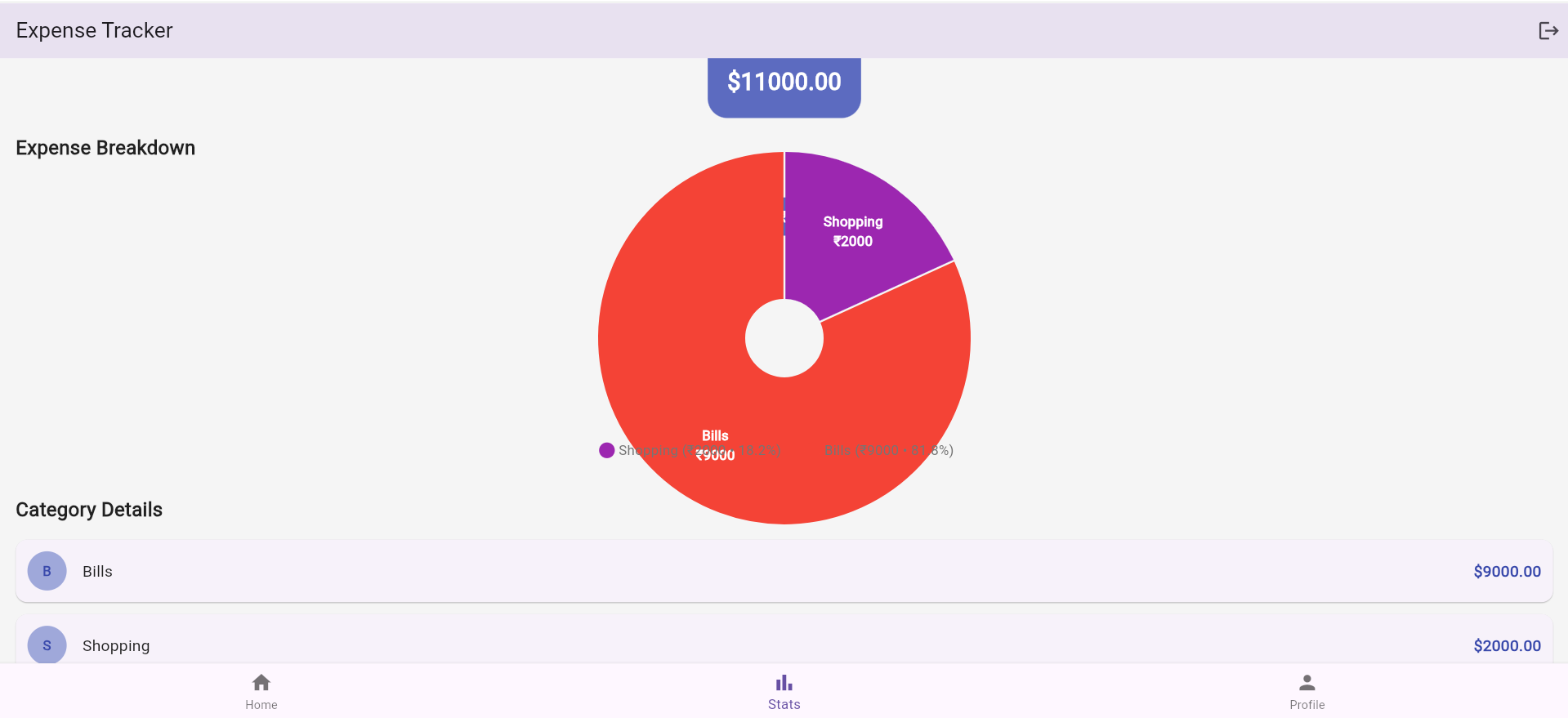
• User taps 'Login' → Home Screen opens.



• User taps '+' button → Add Expense screen is shown.



User taps on the "Stats" tab → A pie chart is displayed showing expense distribution by category.



## CONCLUSION:

This experiment demonstrated how to implement navigation, routing, and gestures in a Flutter app. In the Expense Tracker app, these features help users move between screens and interact with the UI efficiently, improving the app’s overall usability and user experience.