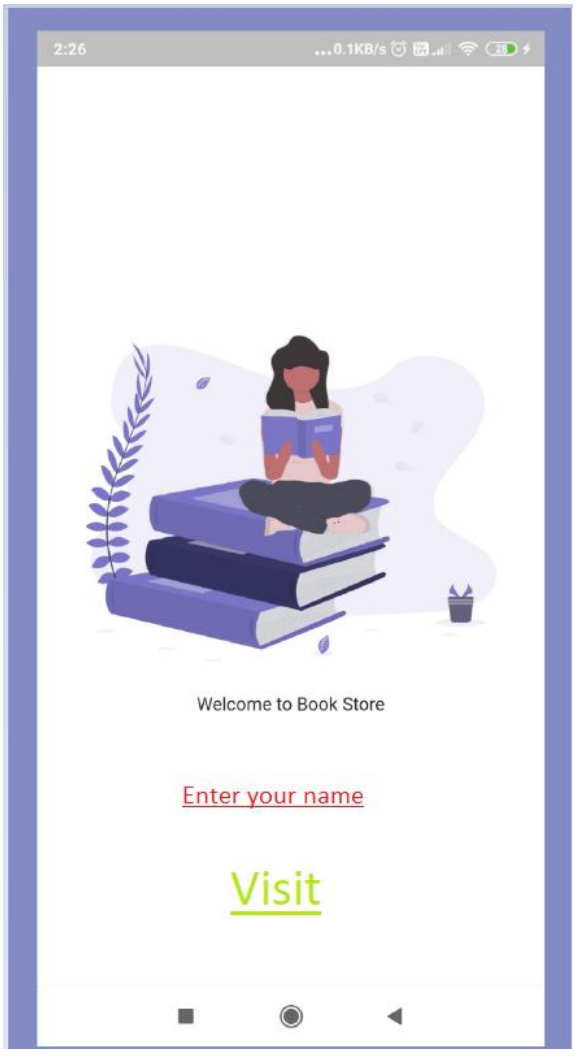
420-431-VA Application Development 2 (Mobile)

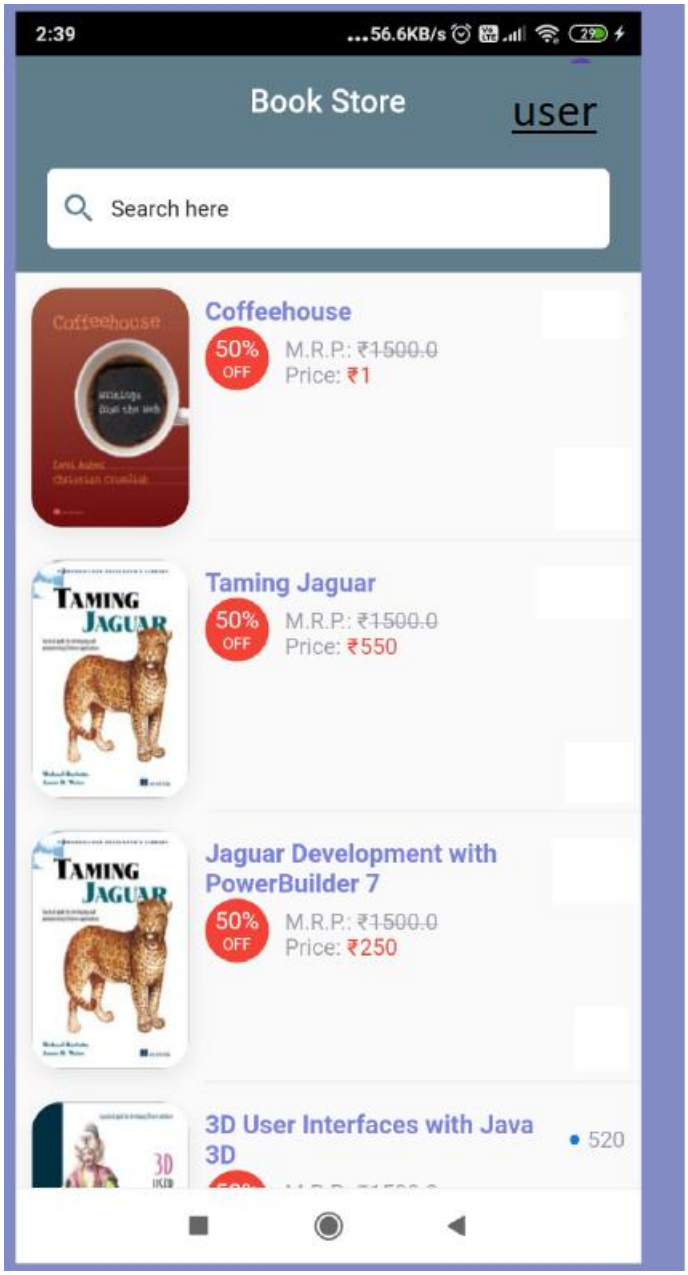
Assignment 1

Sakkaravarthi Ramanathan

Task1) To create and implement a Book reservation application, you need to design four screens, each with specific instructions. Screen 1: This screen enables users to access the first page in which there is a picture, a **textedit to accept the user's name** and a button 'visit' to navigate the next page. Upon clicking the button **visit**, the app should navigate to the second screen while displaying the user's entered name on subsequent screens.

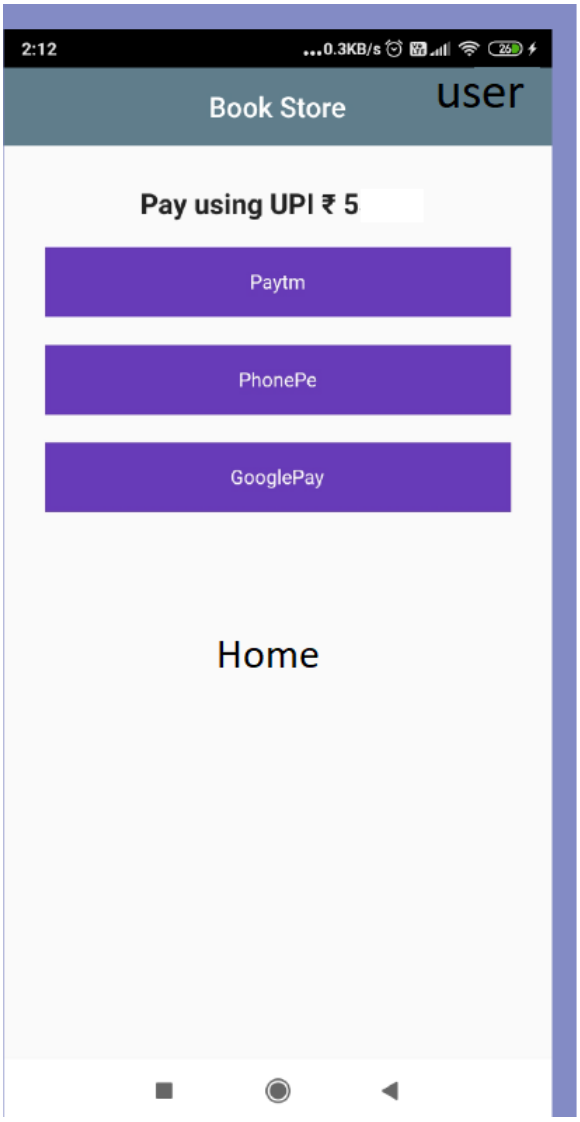


Screen 2: Here, users will enter the main interface, presenting a list view of images, names, and prices of books (at least three books will be displayed). Upon clicking on a book image, users will be directed to a subsequent screen for a thorough description (at least two books will be clickable). Additionally, the text 'User' positioned at the top will dynamically showcase the name entered on the preceding page.



Screen 3: Users can access this page by tapping the picture on the previous screen. Here, they can view the detailed description and button to buy this book as depicted below. By clicking the + and – buttons, the users are opting to select the number of copies and the price should change

Screen 4: After the user chooses the number of copies and taps the buy button on screen 3, the application will present the price along with relevant payment options. If the user opts for Paytm, a snackbar should appear, expressing “thanks for the payment of 5 dollars, your request has been processed”. Furthermore, a "home" button will be situated at the bottom of the screen for easy navigation back to Screen 2.



Dart tasks:

1) Write a Dart program that takes a list of strings and uses an anonymous function to find and **print the longest string in the list.**

   Input : [apple, banana, orange, grapes, mango]

   Output: banana

2) Write a Dart program called sumOfSquares that takes a list of integers and another **function as a parameter**. This function should apply the provided function to each element of the list and return the sum of the squares of the results.

   Input : [1 ,2,3,4,5]

   Output : 55

3) Define a Person class in Dart with instance variables name and age. Define a **named constructor Person.fromNameAndAge(String name, int age)** that takes a name and an age as parameters and initializes the Person object with the provided values. In the main, create an instance for Person and access this constructor
4) Create a Dart program called sum that takes an **optional positional parameter** numbers, which is a list of integers. If numbers are provided, the function should return the sum of all the integers in the list. If numbers are not provided, the function should return 0.
5) Create a Dart program called calculateArea that calculates the area of a shape. The function should have two **optional named arguments**: length and width. If both length and width are provided, the function should return the area of a rectangle (length * width). If only length is provided, the function should return the area of a square (length * length). If neither length nor width is provided, the function should return 0.
6) Create a Dart program called delayedPrint that takes a string message and an integer delay time in milliseconds as parameters. Inside the function, use **Future.delayed** along with async/await to print the message after the specified delay time.

7) Define a Dart class named BankAccount with instance variables accountNumber, accountHolderName, and balance. Implement a method named deposit that takes an amount as a parameter and adds it to the balance. Implement another method named withdraw that takes an amount as a parameter and deducts it from the balance if sufficient funds are available.
8) Create a Dart function called findUniqueElements that takes a list of integers as input and returns a list containing only the unique elements from the input list. If an element appears more than once in the input list, include it only once in the output list.
   Example Input: [1, 2, 3, 2, 4, 5, 3, 6] Expected Output: [1, 2, 3, 4, 5, 6]
9) Implement a Dart function named isPalindrome that takes a string as input and returns true if the string is a palindrome (reads the same forwards and backwards), and false otherwise.
   Example Input: "radar" Expected Output: true.
10) Write a Dart program that uses a map to store the names and salary of a group of people. Then, the print the names of people whose salary are in between 50000 to 75000

Note:

- Take the initiative to understand the tasks independently. If you encounter any doubts, try to resolve them on your own. Avoid asking the teacher about every single question.
- After finishing the task, organize the APK file and all related Dart files into a folder. Then, submit the folder to the Assignment link. Additionally, it's recommended to demonstrate your completed Flutter Task in the lab itself.

****