**MASTER OF COMPUTER APPLICATIONS**

**MOBILE APPLICATION DEVELOPMENT LABORATORY**

**EXPERIMENTS**

**Experiment-1)**

You are assigned to create a book application in React Native that employs stack navigation. The app consists of two screens: the first screen gathers form details about a book (e.g., title, author, genre), and the second screen displays these form entries upon submission.

* Explain how to setup a React Native Application
* Enumerate the components and form fields you would use to gather information such as book title, author, and genre.
* Implement the idea of stack navigation to your application
* Discuss the specific hover animation you intend to apply to the submit button

**Experiment-2)**

You are tasked with developing a blog-like application in React Native, where each blog post consists of an image, title, and description. Additionally, each blog post has a "Like" button that, when clicked, changes its color, and the like count is incremented by 1.

* Explain how to setup a React Native Application
* Describe how you would structure the blog post component to display an image, title, description, and the "Like" button.
* Discuss the React Native styling techniques to change the color of the "Like" button upon clicking it.
* Explain how to incorporate animation (e.g., fade or color transition) to create a smooth visual effect when the button changes color.

**Experiment-3)**

Build a basic React Native weather application that retrieves and displays the current temperature and weather condition for a predefined location on clicking a button. Use a weather API of your choice to fetch the data.

* Describe the initial steps to set up a new React Native project for the weather application. And, explain how to install any necessary dependencies.
* Choose a weather API and explain how to make an API request to fetch weather data for a specific location.
* Change the background image of the application
* Explain how to incorporate animation (e.g., fade or color transition) to create a smooth visual effect when the button changes color.

**Experiment-4)**

Create a basic React Native application that features tab navigation. The app should have two tabs: one displaying a list of popular movies, and the other showing a list of favourite TV shows.

* Describe the initial steps to set up a new React Native project for the weather application. And, explain how to install any necessary dependencies.
* Explain how you would implement this tab navigation and the essential components and steps involved.
* Toggle between dark and light mode
* Add transition effects to the application

**Experiment-5)**

Develop a React Native application that features a digital clock displaying the current time, with the ability to switch between dark and light themes. Additionally, implement a timer with popup features for the application.

* Describe the initial steps to set up a new React Native project
* Describe how you would create a digital clock that displays the current time in hours, minutes, and seconds format.
* Explain how to implement dark and light theme functionality in the React Native application.
* Explain how to use components like Modal to display the timer popup.

**Experiment-6)**

Design a React Native application that includes a feedback form with specific fields and validation, event handling for form submission and clearing.

* Describe how to create a feedback form in React Native with the following fields: Name, Email, Course, and Rating.
* Explain how to handle the OnSubmit event to validate the form data before submission. Describe the logic for checking that all required fields are filled and that email validation is successful.
* Discuss how to handle the OnPress event when the user clicks a "Clear" button to reset the form fields.
* Explain how to set a maximum and minimum character limit for the Name field to ensure it falls within an acceptable range.
* Explain how to validate the user's rating choice to ensure it falls within a specified range (e.g., 1-5). Describe how to handle situations where a user attempts to submit a rating outside the allowed range.

**Experiment-7)**

Create a React Native application that includes a login form with email and password fields, a background image, the ability to display a modal/alert box with a message upon button click, and immediate generation of a notification when the user's location is retrieved.

* Describe how to create a login form in React Native with fields for email, password, and a login button.
* Describe how to display a modal or alert box with a custom message when the button is clicked.
* Explain how to retrieve the user's location in a React Native application. Describe how to generate a notification immediately when the user's location is retrieved.
* Detail how to set a background image for the React Native application. Discuss styling considerations to ensure the background image fits the screen.

**Experiment-8)**

Design a React Native application that serves as a BMI (Body Mass Index) calculator. The app should include a background image, stack navigation for multiple screens, and a modal or alert box to display the calculated BMI result.

* Describe the initial steps to set up a new React Native project for the BMI calculator. Explain how to configure stack navigation to accommodate multiple screens.
* Outline the components and layout required for the app's user interface, including input fields for height and weight. Discuss how to design a visually appealing input form for capturing user data.
* Describe the logic for calculating the BMI based on user input (BMI formula: weight (kg) / (height (m) \* height (m))). Explain how to display the calculated BMI result on a separate screen or in a modal/alert box.
* Explain how to set a background image for the React Native application. Describe styling considerations to ensure the background image fits the screen and complements the overall design.

**Experiment-9)**

Develop a React Native application for a temperature converter calculator. The app should and display the result using a modal/alert box. The application should also feature a background image and stack navigation for a seamless user experience.

* Describe the initial steps to set up a new React Native project for the temperature converter calculator. Explain how to implement stack navigation to navigate between different screens within the app.
* Outline the components and layout required for the app's user interface, including input fields for entering the temperature value and selecting the unit (Celsius or Fahrenheit). Explain how to create user-friendly input forms for temperature conversion.
* Describe how to set up and style the background image for the application.
* Describe how to use a modal or alert box to display the converted temperature result to the user. Explain how to trigger the modal/alert box to appear with the converted temperature value.

**Experiment-10)**

Design a React Native application that uses tab navigation to display a grocery list with three screens: "To Buy," "Bought," and "To Avoid." The list items should be categorized accordingly. Additionally, include a feature to toggle between dark and white themes.

* Describe the initial steps to set up a React Native project with tab navigation. Explain how to create three screens ("To Buy," "Bought," and "To Avoid") for displaying grocery list items.
* Outline the components and layout required to display the grocery list items, categorized as "To Buy," "Bought," and "To Avoid." Explain how to use FlatList or SectionList to efficiently render the list items with appropriate categories.
* Detail the implementation of a theme toggle feature that allows users to switch between dark and white themes.
* Add custom Google font to the application

**Experiment-11)**

You are tasked with developing a React Native application for college admission registration. The app should provide a user-friendly experience with tab-based navigation for different screens, a custom bottom tab bar with animations, an admission form with necessary fields, validation upon form submission, and attractive transitions and animations.

* Describe the initial steps to set up a React Native project with tab-based navigation for the college admission registration application. Explain how to configure and create screens for the different tabs.
* Detail how to create an admission form with all the necessary fields to collect applicant information. Enumerate the essential fields typically included in a college admission form (e.g., name, contact information, academic history).
* Describe the process of performing client-side validation on the admission form to ensure all required fields are filled and correct data is entered.
* Discuss the importance of applying transitions and animations to enhance the user experience and provide visual feedback.

**Experiment-12)**

Create a simple React Native application for a product purchase payment calculator. The app should allow users to enter the product price, select a payment method (e.g., cash, credit card), and calculate the total amount to pay based on the selected payment method.

* Outline the components and layout required to create the user interface for entering the product price and selecting a payment method. Describe how to incorporate input fields, dropdowns, or radio buttons for user interaction.
* Explain the formula and calculations needed to calculate the total amount.
* Describe the content and format of the modal/alert box, including details such as product price, selected payment method, and total amount.
* Describe how to implement a dark and light theme toggle feature that allows users to switch between different app themes.

**Experiment-13)**

Create a React Native application for a to-do list. The app should allow users to add, edit, and remove tasks. Additionally, implement a feature to categorize tasks into different lists, and provide the ability to mark tasks as complete.

* Outline the components and layout required to create the user interface for adding, editing, and removing tasks. Describe how to incorporate input fields, buttons (add, edit, delete), and lists of tasks.
* Explain how to display tasks in a user-friendly format, including task names and completion status.
* Describe how to implement a dark mode feature that allows users to switch between light and dark themes for the application.
* Add hover transitions

**Experiment-14)**

Develop a React Native application that features a digital clock displaying the current time in 24 hrs, with the ability to switch between dark and light themes. Additionally, implement a timer with popup features for the application.

* Describe the initial steps to set up a new React Native project
* Describe how you would create a digital clock that displays the current time in hours, minutes, and seconds format.
* Explain how to implement dark and light theme functionality in the React Native application.
* Explain how to use components like Modal to display the timer popup.

**Experiment-15)**

Create a React Native countdown timer application that starts from 10 and counts down to 0. When it reaches 0, trigger an alert box with a message. Additionally, implement a theme toggle feature that allows users to switch between light and dark themes. Apply smooth transitions for an enhanced user experience.

* Outline the components and layout required to create the user interface for the countdown timer. Describe how to incorporate a timer display, countdown control buttons, and the alert box trigger.
* Describe how to create a mechanism that triggers an alert box with a specific message when the countdown timer reaches 0. Explain how to customize the alert box message and appearance.
* Explain how to implement a theme toggle feature that allows users to switch between light and dark themes for the application.
* Describe how to incorporate transitions

**Experiment-16)**

You are assigned to create a student detail management application in React Native that employs stack navigation. The app consists of two screens: the first screen gathers form details about the students, and the second screen displays these form entries upon submission.

* Explain how to setup a React Native Application
* Enumerate the components and form fields you would use to gather information such as student name, class, age, register number, and so on.
* Implement the idea of stack navigation to your application
* Discuss the specific hover animation you intend to apply to the submit button