Project Definition: Weather App (Sky Scrapper)

Project Description:

"Sky Scrapper" is a feature-rich weather application built using Flutter, incorporating key concepts such as API calling, Shared Preferences, and network connectivity. The app offers a comprehensive set of functionalities designed to provide users with accurate and timely weather information for cities, states, and countries.

"Sky Scrapper" aims to be a reliable and user-friendly weather application, empowering users with accurate and timely weather information. By incorporating API calling, Shared Preferences, network connectivity checks, city/state/country-wise weather details, saved location functionality, light and dark themes, splash screen, and detailed weather screens, the app provides a comprehensive solution for users to stay informed about weather conditions in their desired locations.

Project Objectives:

API Calling: The app utilizes API calls to fetch weather data from reliable weather data providers. This ensures that users receive accurate and up-to-date weather information for their desired locations.

Shared Preferences: "Sky Scrapper" incorporates Shared Preferences to enhance the user experience. Users can save specific cities or states of interest, and this information is stored locally. The app provides a dedicated page where users can view their saved locations for quick access to weather details.

Network Connectivity: The app includes network connectivity checks to ensure uninterrupted access to weather data. In scenarios where network connectivity is unavailable, the app gracefully handles errors and presents an error screen, notifying the user about the network unavailability and suggesting alternative actions.

Fetch City/State/Country Wise Weather Detail: Users can search for weather details by entering the name of a city, state, or country. The app retrieves the relevant weather information and displays it in an organized and user-friendly manner, including temperature, humidity, wind speed, and other relevant meteorological data.

Light & Dark Themes: "Sky Scrapper" offers a visually pleasing experience with customizable themes. Users can switch between light and dark themes based on their preferences, ensuring comfortable usage in different lighting conditions and personalizing their weather app experience.

Splash Screen: The app features an appealing splash screen that creates a positive first impression on users. The splash screen can showcase the app logo, weather-related imagery, or a visually engaging animation, captivating users and setting the tone for their weather exploration.

Detail Screen for Each City/State Wise Weather Detail: Each city or state's weather information is presented on a dedicated detail screen. This screen provides a comprehensive overview of the current weather conditions, along with additional details such as hourly and daily forecasts, precipitation chances, and sunrise/sunset times. Users can explore and analyze the weather data in depth for the selected location.

Project Technologies:

- Dart language
- Flutter framework with MVC and Provider architecture

Project Requirements:

- Knowledge of programming languages such as Dart and JavaScript.
- Familiarity with the Flutter framework, API Calling, and JSON parsing.
- Understanding of user interface design principles and best practices.
- Experience with mobile app development and related tools like Android Studio or VS Code.
- Ability to work independently.

Project Deliverables:

- A functional Weather application developed using Flutter.
- Source code and documentation for the application.
- User manual and installation instructions.
- Project report detailing the development process, challenges faced, and solutions implemented.
- Presentation and demonstration of the application to a panel of reviewers.

Project Evaluation:

- User satisfaction with the application's functionality and UI/UX
- Quality of code and project structure
- Presentation quality and clarity of communication

Instructions:

Task Completion: Ensure that you attempt all the assigned tasks given to you as part of the exam. Complete each task to the best of your ability, following the instructions provided.

Assumptions: Make suitable assumptions wherever necessary, based on the requirements and instructions provided. Clearly document any assumptions made in your project documentation or README.md file.

GitHub Repository: Create a GitHub repository to host your project. Upload your project files, including source code, documentation, screenshots, GIFs, and APK file to the repository. Ensure that you provide a clear and descriptive README.md file that includes screenshots and a GIF of the output of your project.

Screenshots and GIFs: Include screenshots and a GIF of the output of your project in the README.md file of your GitHub repository. Screenshots and GIFs clearly demonstrate the functionalities and features of your project, as per the requirements provided.

Professional Manner: Maintain a professional manner in your project documentation, README.md file, and any other communication related to the exam. Use clear and concise language, follow proper formatting and documentation standards, and avoid any form of plagiarism.

No Copying: Do not copy code, any other content from your classmates, or any other source. Plagiarism is strictly prohibited and can result in severe consequences, including academic penalties. Ensure that all the code and content in your project are original and properly attributed to the appropriate sources, if applicable.

Submission: Once you have completed your project and uploaded it to your GitHub repository, submit the GitHub repository link to your instructor or as instructed. Double-check that your repository is properly organized and includes all the required files, screenshots, GIFs, and APK files as per the instructions provided.

Remember to follow the instructions provided in a professional manner, make suitable assumptions wherever necessary, and avoid copying code or content from any unauthorized sources. Good luck with your project work!

Weather App (Sky Scrapper)
FLUTTER

BRING ON YOUR CODING ATTITUDE