Index

Introduction

The thirst for learning, upgrading technical skills and applying the concepts in real life environment at a fast pace is what the industry demands from IT professionals today. However busy work schedules, far-flung locations, unavailability of convenient time-slots pose as major barriers when it comes to applying the concepts into realism. And hence the need to look out for alternative means of implementation in the form of laddered approach.

The above truly pose as constraints especially for our students too! With their busy schedules, it is indeed difficult for our students to keep up with the genuine and constant need for integrated application which can be seen live especially so in the field of IT education where technology can change on the spur of a moment. *Well, technology does come to our rescue at such times!!*

Keeping the above in mind and in tune with our constant endeavour to use Technology in our training model, we at Aptech have thought of revolutionizing the way our students learn and implement the concepts using tools themselves by providing a *live and synchronous eProject learning environment!*

So what is this eProject?

eProject is a step by step learning environment that closely simulates the classroom and Lab based learning environment into actual implementation. It is a project implementation at your fingertips!! An electronic, live juncture on the machine that allows you to

- Practice step by step i.e. laddered approach.
- Build a larger more robust application.
- Usage of certain utilities in applications designed by user.
- Single program to unified code leading to a complete application.
- Learn implementation of concepts in a phased manner.
- Enhance skills and add value.
- Work on real life projects.
- Give a real life scenario and help to create applications more complicated and useful.
- Mentoring through email support.

The students at the centre are expected to complete this eProject and send complete documentation with source code to eProjects Team

Looking forward to a positive response from your end!!

Objectives of the project

The Objective of this program is to give a sample project to work on real life projects. These applications help you build a larger more robust application.

The objective is not to teach you the concepts but to provide you with a real life scenario and help you create applications using the tools.

You can revise them before you start with the project.

It is very essential that a student has a clear understanding of the subject.

Kindly get back to eProjects Team, in case of any doubts regarding the application or its objectives.

Background:

The City Guide mobile app is a digital platform designed to provide residents and tourists with valuable information and recommendations for exploring and enjoying a specific city. It serves as a user-friendly, all-in-one solution for discovering local attractions, restaurants, events, accommodations, and more. The app leverages the power of mobile technology to enhance the city exploration experience and make it more accessible to a wide range of users.

Functional Requirements-

User Registration and Authentication:

a. Users can create accounts.
b. Users can log in securely to access personalized features and settings.
c. Password reset option should be available.

City Selection:

a. Users can browse and select a city from a list of available cities.
b. The app provides information about each city, including descriptions and images.

Attraction Listings:

a. The app displays a list of popular attractions, restaurants, hotels, and events in the selected city.
b. Users can filter and sort attraction listings based on categories and ratings.
c. Each listing includes the attraction's name, image, description, contact information, opening hours, and user ratings.

Detailed Information:

a. Users can view detailed information about each attraction by tapping on a listing.
b. Detailed information includes additional images, location on the map, user reviews, and a link to the attraction's website.

Maps and Directions:

a. The app provides an integrated map with markers for attractions.
b. Users can get directions to an attraction from their current location.

User Reviews and Ratings:

a. Users can leave reviews and ratings for attractions they have visited.
b. Reviews should include text comments and star ratings.
c. Users can like helpful reviews.

Search Functionality:

a. Users can search for specific attractions, restaurants, or events by name or keywords.
b. The app provides search filters to help users find what they're looking for.

User Profile and Preferences:

a. Users can view and edit their profiles, including their name, profile picture, and contact information.
b. Users can set their preferences, including their favorite attractions and notification settings.

Admin Dashboard:
a. An admin dashboard should be available to manage attraction listings, reviews, and notifications.
b. Admins can add, edit, or remove attractions, events, and other content.

Non-Functional Requirements-

Responsiveness: The app should respond to user interactions within 1-2 seconds, ensuring a smooth and lag-free experience.
Loading Time: The app's initial loading time should be minimized to ensure users can access it quickly.
User Interface: The app's user interface should be intuitive, following best design practices for mobile apps to ensure ease of use.
Accessible: The application should have clear and legible fonts, user-interface elements, and navigation elements.
User-friendly: The application should be easy to navigate with clear menus and other elements and easy to understand.
Operability: The application should operate in a reliably efficient manner.
Error Handling: Implement robust error handling to provide clear error messages to users and gracefully handle unexpected situations.
Scalability: The application architecture and infrastructure should be designed to handle increasing user traffic, data storage, and feature expansions.
Security: The application should implement adequate security measures such as authentication. For example, only registered users can access certain features.
User Documentation: Provide user guides, FAQs, and tutorials to help users understand and navigate the application.
Developer Documentation: Maintain developer documentation to assist in further development and maintenance.
Video: Provide video displaying complete working of the application.

Hardware/ Software Requirements

Hardware

- A minimum computer system that will help you access all the tools in the courses is a Pentium 166 or better
- 128 Megabytes of RAM or better
- Windows 2000 Server (or higher if possible)

Software

Use software as per your requirement

- Windows OS/JAVA/Android SDK/Notepad/SQL/Dart/Flutter