Nokwazi Kunene

Final Project Plan Class Diagram

This project aims to help choose the perfect vacation spot in South Africa based on preferences that are inputted and the current weather conditions.

+------------------+

| Location |

+------------------+

| - Name: string |

| - Region: string |

| - Weather: string|

+------------------+

| + DisplayInformation(): void |

+------------------+

^

|

+---------------+

| City |

+---------------+

| - Currency: string |

+---------------+

| + DisplayInformation(): void |

+---------------+

^

|

+---------------+

| Beach |

+---------------+

| - WaterTemperature: string |

+---------------+

| + DisplayInformation(): void |

+---------------+

+-----------------+

| OutfitAdvisor |

+-----------------+

| + GetOutfit(weather: string): string |

+-----------------+

+-----------------+

| ActivityAdvisor |

+-----------------+

| + GetActivities(region: string): List<string> |

+-----------------+

+-------------------+

| RestaurantAdvisor |

+-------------------+

| + GetRestaurants(region: string): List<string> |

|  |
| --- |
| Program  ------------------  + Main(args: string[]): void  ------------------ |

The City class and Beach class are derived from the Location class, adding specific properties like currency and water temperature, respectively.

They are responsible for suggesting outfits based on weather, suggesting activities based on the region, and suggesting restaurants based on the region, respectively. Each utility class provides a public method for retrieving the suggestions.

The above class diagram represents the classes involved in the vacation advisor program. The Location class is the base abstract class with attributes such as Name, Region, and Weather. It also has a method DisplayInformation() to display the details of a location.

The City class is a derived class from Location and adds an additional attribute, Currency, which represents the currency used in the city.

The Beach class is another derived class from Location and includes an attribute, WaterTemperature, to represent the temperature of the water at the beach. They also override the DisplayInformation() method to display the additional information specific to cities and beaches.

The OutfitAdvisor, ActivityAdvisor, and RestaurantAdvisor classes are utility classes that provide specific functionalities.

The OutfitAdvisor class has a single static method, GetOutfit(), which takes the weather as input and returns a suggested outfit based on the weather condition.

The ActivityAdvisor class also has a static method, GetActivities(), which takes the region as input and returns a list of suggested activities to do in that region.

The RestaurantAdvisor class provides a static method, GetRestaurants(), which takes the region as input and returns a list of recommended restaurants to try in that region.

The OutfitAdvisor, ActivityAdvisor, and RestaurantAdvisor classes are responsible for suggesting outfits based on weather, suggesting activities based on the region, and suggesting restaurants based on the region, respectively. Each utility class provides a public method for retrieving the suggestions.

Finally, the Program class contains the Main() method, which serves as the entry point of the program.