



CODEKRAKEN PROJECT

AMBULANCE TRACKER USING PYTHON

GROUP CONSISTS OF -

- 1. MANAV SHAH**
- 2. JAINEEL JOSHI**
- 3. MANAV GOYAL**
- 4. ADITYA PHATANGARE**

ABOUT OUR PROBLEM

In today's traffic world, ambulance plays a major role when an accident occurs on the road network and the need arises to save valuable human life.

Transportation of a patient to an emergency hospital seems quite simple but in actuality, it is quite difficult and gets more difficult during peak hours.

When an ambulance doesn't arrive on time, it can have serious consequences, as timely medical care is critical in many emergency situations.



SOLUTION

In our Ambulance Booking/tracking System based on C++ programming Language, people can easily book or track an ambulance. There are three major modules namely User, Ambulance, and Hospital. Users can register and log in using credentials. Users can edit their profile and change their password in an emergency. Any Upcoming Ambulance Booking details if anyone wants to Book an Ambulance or if there is an Emergency.

FRAMEWORK

Explanation:

- This section includes necessary header files (`iostream`, `vector`, `ctime`, `cstdlib`) and defines a namespace (using namespace `std`) for input/output operations.
- A struct named `Ambulance` is defined to hold information about ambulances. It has three attributes: `name`, `hospital`, and `contact`.
- The `User` class is created to represent user profiles. It has a `username`, `password`, and a vector of bookings. The constructor initializes a user object with a username and password.
- `findNearestAmbulance` is a function that simulates finding the nearest ambulance. It returns a randomly selected ambulance from a list of simulated ambulances.
- `registerUser` allows users to create profiles by entering a username and password. It checks if the username is unique among existing users and registers the user if it is.
- `login` allows users to log in by entering their username and password. It verifies the credentials against the user profiles and returns the user object if the credentials are valid.
- `bookAmbulance` is used to book an ambulance.
- It collects information about ambulance size, pickup point, hospital, date, and time from the user.
- The nearest ambulance is found using `findNearestAmbulance`.
- Booking details and the information of the nearest ambulance are displayed to the user.
- The booking is recorded for the user.

FRAMEWORK

Explanation:

- `viewBookings` is used to display the bookings made by a user.
- It iterates through the user's bookings and prints the ambulance size and pickup point for each booking.
- The main function is the entry point of the program.
- It seeds the random number generator with the current time.
- It maintains a vector of user data (`users`) and a pointer to the currently logged-in user (`currentUser`).
- The main loop displays a menu to the user, where they can choose options like registration, login, booking an ambulance, viewing bookings, or exiting the system.
- The loop continues until the user chooses to exit the program.

UNIQUENESS

- The uniqueness of an ambulance booking system idea can come from various aspects of its design, features, and implementation.
- The uniqueness of an ambulance booking system idea often lies in its ability to address specific challenges or gaps in emergency medical services, incorporate cutting-edge technology, and cater to the unique needs of the community it serves. Conduct thorough market research, gather feedback from potential users, and explore innovative solutions to set your ambulance booking system apart from others in the field.