**PROJECT REPORT:**



**Project Name:** HospiTech

**Made By:**

21k-3881 (Muhammad Taha Jawaid)

**Problem Statement:**

For patients who want to visit our hospital and want a checkup, HospiTech allows patients an automated appointment system where they can choose their required doctor according to their fee and education, we have multiple doctors of the same disease of d/f education and fee if the doctor is only MBBS so his fee is low as compared to the doctor whose education is MBBS+FCPS so his fee is high and we also set the priority of the patient that if the patient is in emergency so automatically the doctor will check him first as of higher priority and the normal patient will be check by the doctor after the emergency patient. We also applied a function in which the doctor will insert the recommendation of each patient onto the software so that the record will be maintained of all the patients. We have done this project because in the hospital only one doctor is available for the specific disease and sometimes the doctor has very high so every one cannot afford that secondly if the patient is in severe condition so he has to wait for his appointment so we give priority to our patient that if he is in emergency condition so he will be checked by the doctor first.

**Description of used Data Structure Concepts:**

We used the following Data Structure concepts:

1. Doubly linked list
2. Priority queue
3. Recursion
4. Merge sort
5. Hashing

We have used the Doubly linked list b/c it allows data to enter in the center of the node, in last, and front as compared to an array that doesn’t allow to insert the data like that. We have used the priority queue b/c it allows us to make any data as of 1st priority so that if the patient is in an emergency he will be our 1st priority. We have used the concept of recursion b/c it allows us to call the function repetitively so that if the doctor has more than one patient he will check the other one after checking the first one. We have used merge sort to sort the data, we applied merge sort in our fee, so that the patient will see the doctor of low fee on first position and similarly of remaining ones. We have used the hashing function

**Time and Space Complexity of used functions:**

The time complexity and space complexity of the used functions are given below:

The time complexity of the doubly linked list is O(1).

The space complexity of a doubly linked list is O(n).

The time complexity of the priority queue is O(log n).

The space complexity of the priority queue is O(n).

The time complexity of recursion is O(n log n).

The space complexity of recursion is O(nm).

The time complexity of the merge sort list is O(n log n).

The space complexity of merge sort is O(n).

The time complexity of hashing is O(n).

The space complexity of hashing is O(n).

**Other Existing Projects:**

There is no similar project, we have applied such unique functions that will be helpful for every category of patients. We make our priority to the patients how much we can help and facilitate to the patients.