**Proposer Details**

| Group Number | *G5* |
| --- | --- |
| Registration Number of Group Members | 2020-CS-108  2020-CS-155 |

**Proposal Details**

|  |  |
| --- | --- |
| ***Project*** |  |
| Proposed Project Title | Hotel Pricing Analytic |
| Executive Summary | Our project named as HOTEL PRICING ANALYTIC, in our project we will make an online system which is used to reserve a room in the hotel and it will show which rooms are available and which are reserved. Also show the location of the hotel ratings and address of the hotel.  In our Project we will scrap the data for our project from different sites including:   1. Booking.com 2. TripAdvisor.com 3. Expedia.com 4. Trivago.com 5. Travelocity.com 6. Hotwire.com   Each website has large amount of information about hotels all over the world  We will use these following entries to scrap like:   1. Hotel Pricing per Room 2. Phone Number 3. Room Type 4. Hotel Reviews 5. Hotel Rating 6. Hotel Address 7. Hotel Location   And also much more as soon as possible according to the needs of project  We will do the code of our Project hotel Booking and Pricing System that describes the functionalities of sorting the data, which we will scrap from the different sites and sort it using different kind of algorithms like Selection Sort, Merge Sort, Quick Sort etc. We will We will do the best efforts in developing this project in a well and different manner using pyqt5 and QT Designer |
| ***Business Case*** |  |
| Outline the business need for the project | Pricing always matter in every kind of business but in our case room pricing is very important when we knows the competition is very high in this field, so we have to make right room pricing. |
| End user of the product | The end user of the product is any type of the person which need a hotel room. |
| Motivation for Project | The Motivation about this Project is that every person who travel outside from the home have the tension in searching hotel room.  So, we want to design a online system which help the people to book any hotel room from any part of the country. |
| State the level of impact expected should the project proceed and implications of not proceeding | The implementation at Operational Level. |
| ***Technical Details*** |  |
| Name of Entity | Hotel Pricing according to room  Hotel Reviews  Hotel Rating  Hotel Room Type  Hotel Address  Hotel Location  Phone Number  And much more |
| Attributes of Entity  (Minimum seven attributes/rows can be increased) | |  |  |  | | --- | --- | --- | | *Name* | *Data Type* | *Description* | | Hotel Pricing | int | We will scrap the pricing of the per room in the hotel | | Hotel Rating | int | We will scrap the rating of the hotel given by the customers so that people can judge which hotel is best from the other | | Hotel Reviews | String | We will scrap the Reviews about the hotel given by the customers so that people can judge which hotel is best from the other | | Hotel Room Type | String | We will scrap the room type mean which room is available and which one is not | | Hotel Location | String | We will scrape the location of hotel where the hotel is placed | | Hotel Address | String | We will scrape the address of hotel where the hotel is located | | Phone Number | String | We will scrape the phone of every hotel according to rules of hotel | |  |  |  | |
| Sample of Scrapping Source | 1. Booking.com   first.png   1. TripAdvisor.com   first.png   1. Expedia.com   second.png   1. Trivago.com   third.png   1. Travelocity.com   four.png   1. Hotwire.com   five.png |
| Github Repository Link |  |
| Sorting Algorithms | Selection Sort, Insertion Sort, Bubble Sort, Merge Sort, Quick Sort ,Shell sort Tree Sort, Counting Sort, Radix Sort, Bucket Sort  inroSort, Cube Sort TrimSort |
| |  |  | | --- | --- | | **Algorithm Name**  **Comparison Algorithm** | **Description(Each algorithm in 2-3 lines)** | | Selection Sort | Selection sort is a simple algorithm in which we select a min number than sort the whole array into another array . It is also comparison algorithm which compares elements. | | Insertion Sort | Insertion sort is simply like a cards in your hand . You pick up one card and placed with another. In insertion sort elements are transferred right at time | | Merge Sort | Merge sort is one of the most efficient sorting algorithm which works on the concept of divide and conquer method. Merge sort divide the list into number of subparts and then call recursively and combine again | | Quick Sort | Quick sort is also a efficient algorithm and is based on the partitioning of array into smaller arrays. Quick Sort partitions the array and then call them recursively. | | Bubble Sort | Bubble sort is comparison algorithm that compares each adjacent pairs of items in a list or array. Than Swapping the items if necessary and printed the sorted list of array. | | Heap Sort | Heap sort is one of the sorting algorithm which is used to arrange a list of elements Order. Heap Sort uses one of the tree concept that called heap tree. | | * Shell Sort | Shell Sort is highly efficient sorting algorithm that is based on insertion sort. This algorithm avoids the large shift as compare the insertion sort algorithm | | * Tree Sort | A tree sort is a sort algorithm that builds a binary search tree from the elements to be sorted and then traverse the tree. | | Hybrid Algorithm |  | | InroSort | Introsort or introspective sort is **a hybrid sorting algorithm** that provides both fast average performance and optimal worst-case performance. Since the three algorithms it uses are comparison sorts, it is also a comparison sort. | | Time Sort | Tim sort is a hybrid stable sorting algorithm derived from merge sort and insertion sort and is used to design to perform well on many kinds of the real world data. | | Non-Comparison Algorithm |  | | Count Sort  Bucket Sort  Radix Sort  Cube Sort | Count sort is based on keys between a specific range. It works by counting the number of objects.  Bucket sort or Bin sort is a sorting algorithm that works by distributing the elements of an array into number of buckets.  Radix sort is an integer sorting algorithm that **sorts data with integer keys** by grouping the keys by individual digits that share the same significant position and value .    Cube sort is a **parallel sorting algorithm** that builds a self-balancing multi-dimensional array from the keys to be sorted. After each key is inserted the cube can be rapidly converted to an array. | | |
| Searching Algorithms | Searching data for an item of a list by using different technique for example searching by names searching by letters and searching by length of different things. Work become much faster if the list is sorted.  We also provide the some kind of searches as   1. Binary Search 2. Linear Search |
| Searching Filters for each data type |  |
| Multi-Level Sorting | From the analysis of various Hotel Discussion Forum websites. We will add many alternatives for client to scraping total page or a specific block or table with assistance of class or Id by getting link from client. |
| Any other features | We will apply and inform you about the features if we apply more in our Project. |
| ***Interfaces for your project*** |  |
| *[Draw layouts in the pencil tool. For each picture of the UI, provide the following table.]*   |  |  |  | | --- | --- | --- | | UI Component Name | Type of UI component | Purpose of UI Component/Other details | | Home | first.png | The UI is a visual and interactive design for the hotel scrapping.  We start our scrapping from home page that consists the drop down list of different pages same like website. | | Scrapping | second.png | Scrapping page consists of two option 1. Sorting  2.Searching  Sorting list and searching list | | About Us | Third.png | About Page tells about constructer of this project. | |  |  |  | |  | Best Of Luck |  | | |