**Proposer Details**

| Group Number | *11* |
| --- | --- |
| Registration Number of Group Members | 2020-CS-61  2020-CS-85 |

**Proposal Details**

|  |  |
| --- | --- |
| ***Project*** |  |
| Proposed Project Title | Scrapping Online Courses |
| Executive Summary | *In this project we are going to scrap the data of online courses available on websites like edx.org, cousera, khan academy and much more. So, the purpose of this project is to scrap all the courses from internet and merge them in the single space so that all can search for their desires under the same roof. We will get you at least the data of one million courses from internet that will include both paid and free courses. These courses will be from world best universities and we will share you the experience of professional institutes and professors. A student can select his course seeing how many numbers of student are already enrolled the course. We will also show you the number of reviews of people on the courses. You can also see which universities are providing these courses. Besides you can see the level of the courses or prerequisite of the courses. I hope this project will help to learn skills and quench the thirst of your knowledge.* |
| ***Business Case*** |  |
| Outline the business need for the project | *This project is helpful in all the domains but specially in the duration of corona lockdown the student body will get high benefit. Students can join free courses and can fill up the deficiency left behind due to online classes.* |
| End user of the product | *The end user of this project will be students. Besides student teachers and professional staff can also get new methods and can update their old concepts.* |
| Motivation for Project | *As I am also a student, we know which difficulties students are facing in the lock down. Where there are lot of issues regarding studies. It is difficult to ask questions about topics and to interact with teachers. So, the basic motivation of this project is to facilitate the student body and make things easy for them. Secondly to serve the Youngs of nation so, they can take part in the prosperity of nation.* |
| State the level of impact expected should the project proceed and implications of not proceeding | *Yes of course this project will have great impact on the education level of our society. Many people are unfamiliar about the correct and effective use of internet. So, with the help of this project we will teach them how they can make their time more productive and efficient. I hope inshallah this project has lot to serve the nation and humanity.* |
| ***Technical Details*** |  |
| Name of Entity | *The entities that we are going to scrap are as follows*   * *Name of university* * *Name of course* * *Students enrolled* * *Number of reviews* * *Level of course* * *Type of course* * *Duration of course / price of course* |
| Attributes of Entity  (Minimum seven attributes/rows can be increased) | |  |  |  | | --- | --- | --- | | *Name* | *Data Type* | *Description* | | nameOfUni /  name of professor | String | This will store name of university offering course. | | nameOfCourse | String | This will store name of course offered by university. | | studentsEnrolled | int | No of students enrolled in course | | reviews | int | Ranking of course | | levelOfCourse /  Duration of course | String | Level of course means  What is its basic level or of intermediate level or of high level | | typeOfCourse | String | Type of course means it is for professional or for specialization | | priceOfcourse | int | Price of course | |  |  |  | |
| Sample of Scrapping Source |  |
| Github Repository Link | *https://github.com/Mumer666/CS261F21PID11.git* |
| Sorting Algorithms | * *Merge Sort* * *Insertion Sort* * *Heap Sort* * *Selection Sort* * *Hybrid Sort* * *Quick Sort* * *Bubble Sort* |
| |  |  | | --- | --- | | **Algorithm Name** | **Description(Each algorithm in 2-3 lines)** | | Merge sort | Merge Sort is a recursive algorithm that call itself until its size reduces to zero. The time complexity of this algorithm is O(nlogn). This algorithm first sort and then merge. | | Insertion sort | This algorithm takes a lit bit of more time than merge sort. It check its next item if it is greater it replace it otherwise remains same. | | Heap sort | Heap sort is a comparison-based sorting technique based on Binary Heap data structure | | Selection Sort | It is a swapping algorithm. It runs iteratively and swap current index with greater index. The time complexity of this algorithm is O(n^2). | | Hybrid Sort | Hybrid algorithm uses two or more sorting algorithms and combine them to sort the problem | | Quick Sort | Quick sort uses low index high index and pivot point to sort the algorithm. This algorithm also takes nlogn time complexity. | | Bubble sort | Bubble sort is also a time taking algorithms it uses two for loop and iteratively take the maximum number to the last | | |
| Searching Algorithms | * Linear search   Linear search is an iterative search algorithm that compares each single element with the whole row. And if it finds that particular element it iterates the count. Otherwise, it compare the next element.   * Binary Search   Binary search is more efficient way of searching and you can say it works intelligently and search the particular item by dividing your array into smaller portions. But for this searching there is a condition that your array must be sorted first. |
| Searching Filters for each data type | We will provide two types of searching filter for string type or for integers type. For string type the user will be able to add filter like search from ‘D’ to ‘E’. and in integer type the user will be able to search all the data from 50 to 100. In this way the user will be able to apply filters on search items. |
| Multi-Level Sorting | We will provide multilevel sorting means you can arrange data in both orders you can arrange either in ascending order or in descending order. We will just use the above sorting algorithms and we will code both types ascending and descending. |
| Any other features | Besides these features we will introduce the time button that will show the time taken by each sorting algorithm. |
| ***Interfaces for your project*** |  |
| |  |  |  | | --- | --- | --- | | UI Component Name | Type of UI component | Purpose of UI Component/Other details | | Table | Table to display rows and columns | We used the table so that we can display the data scrapped from the website. And we could perform operations on it. | | Buttons | Buttons | We used a lot of buttons in our UI these include starting button pausing button. A button that will sort the list of entities. And some buttons so we could switch from one page to the other. | | Combo Box | Combo Box | We used combo box to select different type of algorithms. Like which particular algorithm you want to sort for sorting purpose. | | Labels | label | We used a lot labels to display some information on the home screen. | |  |  |  | | |