CSE215 Object Oriented Programming Midterm and Final Project
2020-2021 Fall

Project Description V1 : 23.11.2020

Project Delivery : End of Finals

Please read this document at least 2 times

Multi-Threaded Robust Responsive Web Crawler Design

There will be no exams for both midterm and finals. Therefore, your score will be 100% given from this project. In this project, you are supposed to develop a desktop-based multi-threaded web crawler using some sort of database. You are not bound to use a certain programming language (you can use c#, java, etc.) but it has to have an interface and has to be a desktop application. Your score will be determined by how many of the necessary features you have achieved to develop.

1. Supporting crawl of multiple websites at the same time
2. An option to be able to set crawl external URLs or not
3. An option to be able to set how many threads will the software use to crawl each website
   a. This is load balancing and very important. Let's say we have 10 different websites to crawl. So the software should be able to start let's say 20 crawling tasks to each website
   b. You need to be able to display how many threads are crawling each website at any given time

4. Using some sort of database to keep track of crawled URLs, waiting to crawl URLs, failed URLs, retry counts, last crawl date, title, description of the crawled URLs, inner text of crawled URLs.

    a. Preferred databases are MSSQL (Developer Edition), Oracle, MYSQL, PostgreSQL, and such

5. No errors should ever crash the software and all of the errors must be handled and logged gracefully

6. The user interface has to be fluent and responsive no matter how many crawling tasks are started

7. Start, pause and stop buttons on the user interface

8. An input box to add new root URLs to crawling task

9. A screen to display running tasks count for each website. This can be like multi-tab or a new WPF screen opened via button click

    a. In this case, it should also show other task statuses. For example for c#: Canceled, Created, Faulted, RanToCompletion, Running, WaitingForActivation, WaitingForChildrenToComplete, WaitingToRun

10. A screen to display latest crawled URLs and errors if happened

11. A screen to display run time, average number of crawled URLs per minute, successfully crawled page count, failed to crawl page count, if HTTP errors happened count of status codes, etc.

12. A button to close the software without any data loss. Same behavior is expected when the user closes the software via X button

13. Software must be able to continue from wherever it left so you have to save software current state and the historical state

14. Three (3) times failed to crawl URLs must be disabled for 24 hours and retried again later

Expected Object Oriented Features

In your software, you are supposed to write given codes below in the respected code blocks. I will search and check the given below codes to see whether you have done the requested feature or not. Several example usage

```csharp
//public class GenericList where T : struct
//2019103015
7 references | Furkan Gözükara, 8 days ago | 1 author, 3 changes
public class GenericList<T>// where T : struct // e.g. if i force
{
    // The nested class is also generic on T.
    //2019103002
    8 references | Furkan Gözükara, 8 days ago | 1 author, 1 change
    private class Node
    {
        // T used in non-generic constructor.
        1 reference | Furkan Gözükara, 8 days ago | 1 author, 1 change
        public Node(T t)
        {
            next = null;
            data = t;
        }

        private Node next;
        3 references | Furkan Gözükara, 8 days ago | 1 author, 1 change
        public Node Next
        {
```

```csharp
//2019103006
3 references | Furkan Gözükara, 8 days ago | 1 author, 2 changes
public T this[int index]
{
    get
    {
        if (index >= Lenght)
        {
            return default(T);//if t is cl
            throw new System.ArgumentExcep
        }
        Node current = head;
        for (int i = 0; i < index; i++)
        {
            current = current.Next;
        }
        return current.Data;
    }
```

```csharp
//fields
//2019103001
double a_width;
double a_length;

// Properties for Length and Width
//2019103001
5 references | Furkan Gözükara, 8 days ago | 1 author, 1 change
public double Width
{
    get
    {
        return a_width;
    }

    set
    {
        a_width = (value < 0) ? -value : value;
        //if (value < 0)
        //    a_width = -value;
        //else
        //    a_width = value;
        //same as above single line
    }
}
```

```csharp
//2019103007
1 reference | Furkan Gözükara, 50 days ago | 1 author, 1 change
public static int myCustomToInt(this string s)
{
    int irOut = 0;
    if (Int32.TryParse(s, out irOut) == true)
        return irOut;
    throw new System.ArgumentException("Given parame
}
```

- Object Oriented Programming (General Approach no specific code)

- Classes and methods must be used for everything (General Approach no specific code)
- No duplicate code, function ever (General Approach no specific code)
- Fields and properties usage in classes (2019103001)
- Default constructor and overloaded constructor usage in classes (Polymorphism) (2019103002)
- Method and operator overloading (Polymorphism) (2019103003)
- Namespace usage example in the project. For example, you can put a class in another namespace and use it in another class (2019103004)
- Both globally and locally exception handling. The errors must be logged properly (2019103005)
- This keyword usage example (2019103006)
- Method extension example with this keyword (2019103007)
- Static and non-static class usage (2019103008)
- Public and private class, variable and method usage (2019103009)
- Class inheritance usage (2019103010)
- Enum usage (2019103011)
- Base and this keyword usage (2019103012)
- Method overriding with virtual methods and class inheritance (Dynamic Polymorphism) (2019103013)
- Abstract class usage (2019103014)
- Generic class usage (2019103015)
- Sealed class usage (2019103016)
- Reference passing vs value passing method usage (2019103017)
- Interfaces usage (2019103018)
- Constant variables (2019103019)

- Dictionary and HashSet (2019103020)

- Structs (2019103021)

- Tuples, Nested Tuples, Tuples with Methods (2019103022)

- Default parameters in methods (2019103023)

- Anonymous Method (2019103024)

- Conditional Operator (2019103025)

- LINQ (2019103026)

- Lock usage (2019103027)

- Concurrent Collections (ConcurrentBag) (2019103028)

- Async and Await (2019103029)

- Using Statement (2019103030)

- Reflections (2019103031)

- Serialization and Deserialization e.g. JSON (2019103032)

- StringBuilder (2019103033)

- Ref keyword in methods (2019103034)

- Yield Keyword (2019103035)

- IEnumerable and IEnumerator (2019103036)

- CultureInfo (2019103037)

- Destructors (2019103038)

- IsNullOrEmpty (2019103039)

- TimeSpan, StopWatch (2019103040)

- Timers (2019103041)

- Events (2019103042)

- Optional Arguments, Named Arguments, and Generic Arguments (2019103043)

- Params Keyword (2019103044)

- Local Functions or Nested Functions (2019103045)

- Delegates and Delegate Multicast and Generic Delegates (2019103046)

- Value Tuples, Value Nested Tuples, Value Tuples with Methods (2019103047)

- KeyValuePair (2019103048)

- NameValueCollection (2019103049)

- Generic List, Generic Dictionary, Generic SortedList, Generic SortedDictionary,
  Generic Stack, Generic Queue (2019103050)

From the above requested features, you need to try to use as many as them in your application. The more you have used, the more points you will get. Every one of them has an abundant amount of examples on the Internet. If you use other than C# language, many of them still should be available in your chosen programming language such as Java. However, their names may change.

General Things That Needs to be Considered

Every project will be controlled one by one. So if you cheat or bring a ready code, Project, you will get FF

There will not a final exam. Your grade will be based on this Project. Thus, take this Project very seriously and start working on it immediately. Look the internet for the parts that you do not know

Write your code with as many as possible explanations. What does do that code block? Of course, put those explanations into your code with comment blocks such as // or /* */

Create an account on https://stackoverflow.com/ and ask there, or look for answers there.

When doing the Project you can email and ask me the parts that you cannot solve.

My email: furkan.gozukara@toros.edu.tr

Also use discord channel for communication

Latest Project delivery date: End of finals but you can also deliver at the butunleme exams

1. **Lastly everyone will record a very detailed video to explain the project.**
2. **Video will be recorded with OBS studio in 1080p (high quality so that I can read explained code easily) resolution.**
3. **It is mandatory to use microphone when recording and explaining the project verbally by speaking.**
4. **Turkish students can record the video in Turkish.**
5. **The project has to be made in English.**
6. **Non-Turkish students can also record their video in Turkish if their Turkish is better than their English. However, they should use English if their English is better than their Turkish.**
7. **When explaining your Project you need to explain every piece of your written code.**
8. **What does that particular code piece or that particular class or that particular method, etc. and how did you write it.**
9. **Moreover, you have to explain and show every feature of your application. So run your application and do example of every**

feature such start crawling, show crawling status, restart, pause, etc.

10. Then after you have to upload your video to the YouTube and set visibility as public or unlisted.

11. Because it has to be accessible.

12. Furthermore add your YouTube video link to the project folder as text.

13. Finally ZIP or RAR your entire Project and upload to your Google drive. After you get your Google drive upload link send me that link as email : furkan.gozukara@toros.edu.tr

14. When sharing your Google drive upload link as email, make sure that your upload's visibility is set correctly. Set it as who has link can view or download so that I can download it.