Computer Selection System According to Requirements

Project Report

Phase III

2020-2021 FALL

Sena Yurtseven   senayurtseven123@gmail.com

Burcu Ağdar   agdarburcu@gmail.com

Emirhan Bilge Bulut   emirhanbilgeblt@gmail.com

<Computer Selection System According to Requirements>

1.   Abstract

We have prepared a list of all the objects we need in the project. -These objects represent classes for the background code, and tables for the database. -We have decided in detail what the properties in these objects will be. This was the first step we took to prepare the Entity-Relationship diagram. It was completed when submitting phase 1 of the report. After submitting the report, we decided exactly which technology would be good for us to use. As a result of this decision, we decided to use .Net Core and depending on it, Microsoft SQL Server. Immediately after these decisions, we prepared a detailed plan. In this plan, we set the time we allocate for the extraction of the necessary data as 3-5 days, and a process of 7 days to create the database. We completed the getting of this data from a site on the internet within 3 days. We had completed the draft version of the ER diagram in 2 weeks. For this reason, we finished editing the draft and creating it in the database within 1 day, and adding the json files to the database as data within 2 days. The queries we wrote to get the accuracy of this data and some statistics of the data matching the results took 1 day. We can say that we have completed the database part while submitting our second report. For this reason, there is only 1 month left for us to run and create the website with the database.

<Computer Selection System According to Requirements>

2.   Completion Report

Our system has reached the desired result in general terms. The computer can do the suggestions correctly. User and Admin part is working correctly. At the beginning of the parts where we waste the most time, it was difficult for us to clean data and work with such big data. The biggest challenge we faced later was the transfer of parameters between views. Here, we used Viewbag or temp data for parameter passing, but after it failed, we switched to session structure. Another difficulty was that we avoided using ready-made templates. We have made an effort to manually write all the User section and css codes. At the beginning of the difficulties that we did not expect in the project, the mistakes we would get due to dealing with this amount of data and the cleaning on this data. Another issue was the lack of documentation. Although there are many resources for asp.net, finding samples and resources for .net core was very difficult. We also had some parameter problems in js checks. In general, the biggest difficulties we did not expect in the project were parameter transition between views, problems due to the large data size, lack of experience for front-end and lack of documentation. There were no problems during the design and design phases. As a solution, we switched parameters between views with ajax and created different views that we did not plan and sent them as a single parameter, keeping multiple parameters that we need to send in those views. Since some data do not match exactly, we used the contains structure because of the large amount of data. Here, interestingly, it was unable to fetch the values that are normally strings, although they are attached to the database, and gave an error due to strange signs. Apart from this, sending the parameter as null in parameter passes was also one of the interesting errors.

<Computer Selection System According to Requirements>
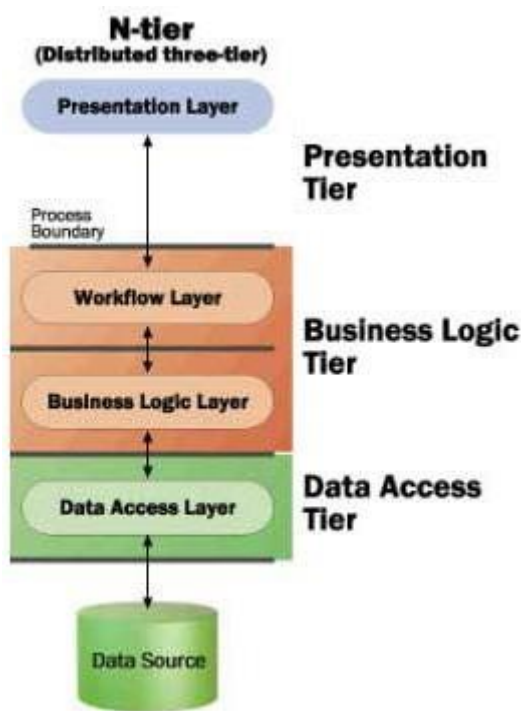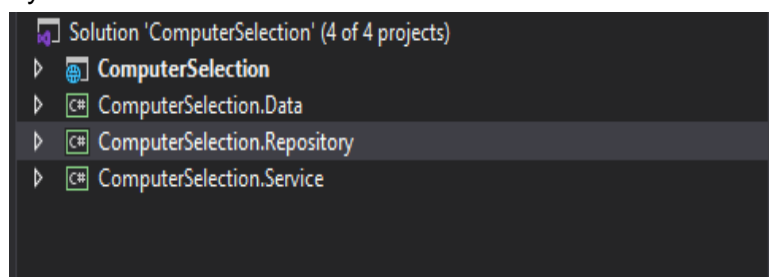
3.   Functional Decomposition

We designed the parts that differ from each other in the project using two different structures. Our first structure is n-tier design and it has a structure like figure (1). According to this structure, we can show the tasks that differ from each other in four layers.


Our first layer is Data Layer the data layer,which is the layer where we store the data .By establishing all entity structures in the code, we can talk about the data we will use when we connect with the database as the layer we create.

Figure 1(N-tier Design)

Our second layer is Bussines Layer and it is divided into two for us. The main reason for dividing into these two will be done jointly.All functions are created in a specific layer with the help of abstract classes and context structures and made ready for use.For example, in a web project, all creation, update, deletion and data listing operations are common. Therefore, in the first area within the Bussines Layer,We have created data operations with the help of a generic T object and we have prepared the functions for which we will perform database operations. Inside the second part of the layer we customized all these processes for each object. We performed database operations that are not common among themselves with the help of basic database operations (CRUD) and thus, we have completed all the functional functions we needed to create.In our last layer, we will interact with the user and using all these functions.

Figure 2(N-tier Design Structure)

<Computer Selection System According to Requirements>
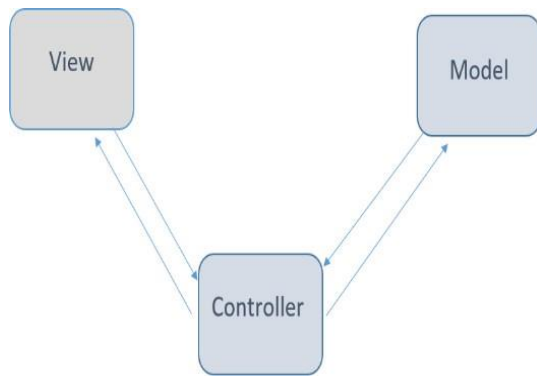
In the last layer, *we* created the area where we can output and receive input. In this area, we used Model-View-Controller (MVC) structure. View in this structure is an input to the user *or* we can create an output field and define it as the first area we will interact with. The model, on the other hand, uses the data we create in other layers. It is the part that creates the inputs or outputs we want to show in views. The structure that provides the connection between these two structures is called the Model. Model also includes two basic functions: Get and Post. Get is the first method to communicate with the user and is the part in which we do not update data. Post is the function that performs what the user wants to do with help of

<Computer Selection System According to Requirements>

4.   High-Level Organization

It has the authority to update manually in the admin database and it has a structure that can directly interfere with adding and removing data. It can access databases and make changes. User records can be accessed or users can be deleted. The Admin panel makes a request to the database via Services. Service structure triggers data retrieval from the Repository layer and the Data layer communicates with the database and fetches the necessary data. The User interacts with the User as soon as he / she enters the website, then directs it to the Laptop or Desktop layer by taking the necessary information from the System Requirement layer. As a result, the User layer sends the data it receives from the SystemRequirement layer in the background, triggering the Dektop or Laptop layers. Laptop or Desktop Layers request Data from the Service layer and send the appropriate data to the User layer. After these data come to the User layer, Laptop and Desktop show them for User. The hierarchy is shown in the figure below. Dashed lines represent temporary hierarchies that indirectly originate from data streams.
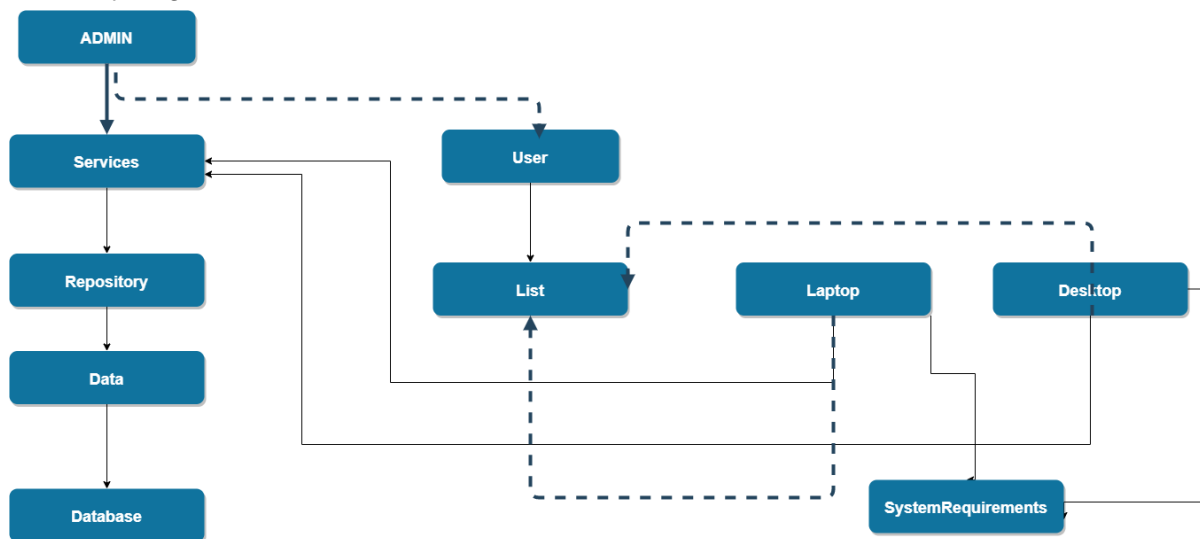
Here, he creates the entire bridge with the Service database, and the User has no direct access to the database as in Admin. Thanks to this hierarchy, the user is disconnected from the database and security has been provided with an n-tier design.

<Computer Selection System According to Requirements>

5.   Clickstreams

Computer selection system is a design that provides convenience to the user. The user will be able to easily find anything he wants thanks to its beautiful design and easy use while logging into the system. The first window that opens as soon as you enter the website gives a brief and concise explanation about the system. It will allow the user to see the system better than the background used. The upper right corner of the first page contains the names of the computer components, computers, system requirements, and admin panel entry. Computer components consist of CPU, GPU, RAM, MAIN CARD, HARDDISK and SSD. When the user clicks on any of these components, they are taken to a page listing all the properties of the components. It is possible to search for certain properties on this page. Similarly, if the user clicks on COMPUTER option, he will be directed to a page listing the computers and their features. It will be able to search on this page according to the specific characteristics of the computers (eg Model, Ram, Processor Model). Another option, SYSTEM REQUIREMENTS, are programs and games the user wants to use. This list contains the system requirements for each program. These requirements are, for example, processors that meet the minimum processor specifications required for a game (shown separately for AMD and Intel) and GPU, RAM, Disk Space requirements.The user will be able to easily find the name of the game he is looking for using the search bar. The user descending from the first page of the website will see two computers under the heading 'Choose your PC'. One is the laptop and the other is the desktop. When the user hovers over the photos, a scrolling list with the program names will be announced below the photos. If the user selects programs from the list under the Notebook photo and clicks the "Create Laptop" button, he will be directed to a table listing the laptops that can support the selected programs.If he selects a program from the list under the desktop photo, there are Cpu and GPUs that support these programs, and cpu, gpu, mainboard ram will be listed. Then the user can select the cpu from the cpu list here and the motherboard list is updated to accommodate those compatible with the selected CPUs. the desktop can access a list of computer components that can support the programs they want. Continuing from the homepage of the website, the user will see summary information about each of the computer components and pictures of these components under the heading "PC Parts". Further down, the computer components recommended by professionals are listed under 'Featured System Recommendations This Month' as entry level, intermediate and high level. Under this section, there are the names of the developers and the emblems of the companies on the site. In addition to all these features for the user on the site, there is also an admin panel for the site owner. The administrator can navigate to this panel from the upper right corner of

<Computer Selection System According to Requirements>

the website. The person who clicks here comes to the administrator login. He will enter his e-mail address and password and enter the admin panel. Once logged in the Admin Panel, the administrator will be able to see the names of all computers, computer components and system requirements on the left system. After clicking any of them, two subtitles will open. For example, after clicking on the cpu text, you will be able to see 'Cpu List' and 'Add Cpu' subtitles, and if the administrator clicks 'Add Cpu', they will be able to add a new CPU. If the administrator clicks on 'List CPU' it will access the list of available CPUs. There are 'Edit' and 'Delete' buttons for each CPU on this list. The user can delete the CPU by clicking the Delete button.If he presses the 'Edit button', he will be able to update existing CPUs.

<Computer Selection System According to Requirements>

6.   Layout

It is the area that meets the screen when the user logs in.
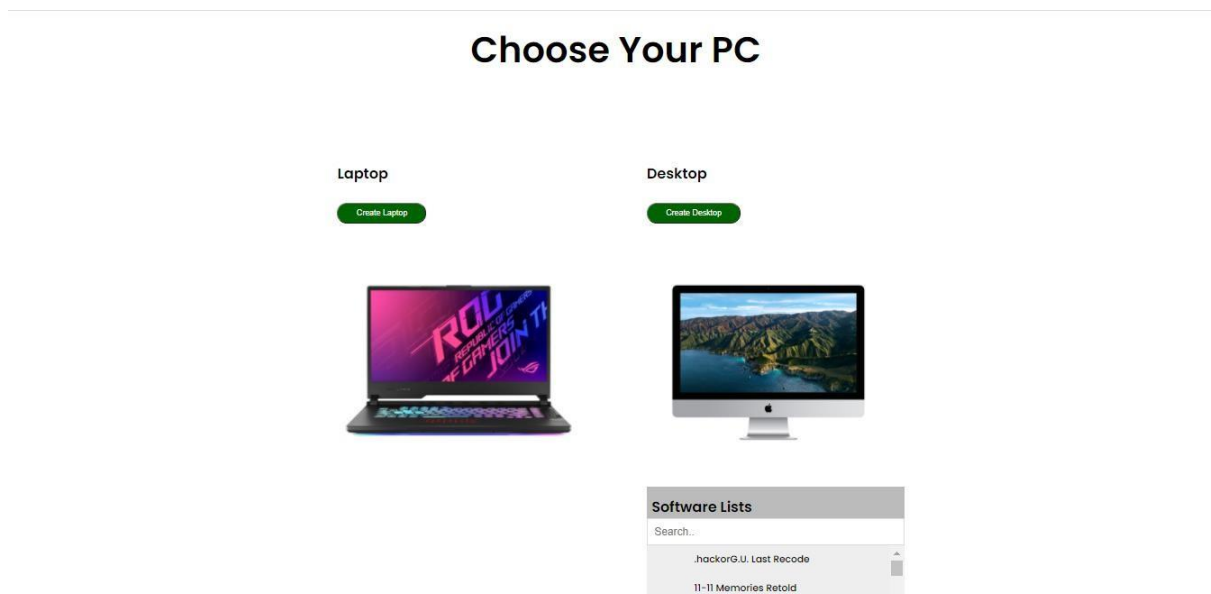
Here, the user has the opportunity to choose the computer or desktop computer he wants. For this, by dragging his mouse on the computer, he chooses the programs that appeared, and the result output is displayed in another window after a while.

<Computer Selection System According to Requirements>
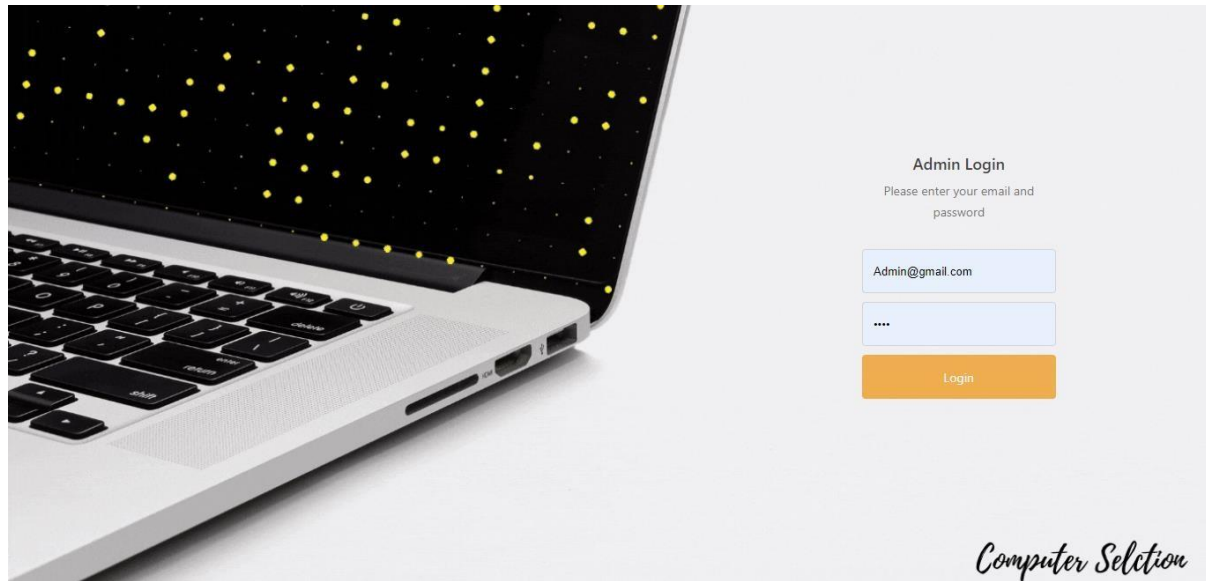
The admin enters the information on the screen he encounters to log in and reaches the panel where the functions he can access are available.

Admin can see the hard drive list with the help of buttons.

<Computer Selection System According to Requirements>

7.   Implementation

The list of computers listed when we make a selection with the help of a computer in the layout section is listed above.

The screen that appears when we log in correctly with the password and user name in the admin panel is above.

<Computer Selection System According to Requirements>

When the admin adds, they can access the output by listing them in the layout section.

<Computer Selection System According to Requirements>

8. Future Work

The size of the data we had was very high and increasing day by day. Due to the size and irregularity of this data, we had to eliminate the data. We were able to use our data potential at 95% in some entities and 40% in other entities. We can do better with these data losses and clearing. Since the data about the old Cpu and Gpu and their official accuracy are not supported by the companies, the structure for the old parts in our recommendations could be improved. The system also prompts not only to select a computer but also for the user 'Will My System Remove This Application? A part in the form of 'can also be included. Not only can the computer be suggested, but also which applications the user can use on the computer he owns can be listed. Since we are trying to write our system as a "responsive website", it will be very easy to make the android mobile application. Android application can be made and adapted. We avoided using ready-made templates and since we write css, html and javascript codes ourselves, improvements in front-end parts, logo designs and color harmony can be increased. Our system is also suitable for providing API services for later developers. After making the necessary arrangements, other developers can easily adapt our system to their own projects through APIs. Cookies and log records can be kept and the user's choices can be remembered to keep and analyze user records and behavior. In general, query optimizations and listing optimizations could be done and the reliability or potential of the system could be tested by applying some penetration tests. Error checks or automatic checks were dynamic in data extraction and insertion, and 'test' applications could be written in different or the same programming languages. The browsers and versions supported by the system could be determined, and optimization could be made for older versions. Data extraction files can be brought into a generic structure by making dynamic or adding functions to the database, and tools can be developed to manage our database.