**THE DOCTOR SEARCH SYSTEM**

**FINAL REPORT**

**CMSE 322**

**PROJECT NO :**

**GROUP NO : 2**

**PROJECT NAME: The Doctor Search System**

**PROJECT START DATE: 01.03.2023.**

**PROJECT END DATE: 29.0.5.2023**

**SUPERVISOR: Assoc. Prof. Dr. Duygu Çelik Ertuğrul**

**SEMESTER TERM: 2022-2023 Spring**

**Computer Engineering Department**

**Eastern Mediterranean University**

**Term: SPRING2022-2023**

# ABSTRACT

The scope of this project is to provide people living in the TRNC with access to information about doctors in the country. Our goal is to provide this coverage in the fastest and most reliable way. While developing the project, we used Web Scraping and Web Crawler methods to provide accurate and up-to-date information. We also integrated the methods accepted in the software community into our project to provide the user with comfortable access to the system we have developed. As a result, we have developed a system where users can search and filter doctors and get detailed information about the doctor.

**Keywords:** Doctor, Doctor Search System, TRNC, Web Crawler, Web Scrapper.

# Table of Contents

[ABSTRACT](#_Toc478723910) II

[Table of Contents](#_Toc478723911) III

[LIST OF FIGURES](#_Toc478723912) V

[LIST OF TABLES](#_Toc478723913) VII

[1. INTRODUCTION 7](#_Toc478723914)

[2. PROJECT PLANNING AND MANAGEMENT 8](#_Toc478723915)

[3. REQUIREMENTS ANALYSIS 3](#_Toc478723916)

[3.1 Functional Requirements 3](#_Toc478723917)

[3.2 Non-Functional Requirements 3](#_Toc478723918)

[3.3 Realistic constraints 3](#_Toc478723919)

[3.4 Ethical issues 3](#_Toc478723920)

[4. DESIGN 4](#_Toc478723921)

[4.1 High level design (architectural) 4](#_Toc478723922)

[4.2 Software design 4](#_Toc478723923)

[5. IMPLEMENTATION 5](#_Toc478723924)

[5.1 Tools, technologies and platforms used 5](#_Toc478723925)

[5.2 Algorithms 5](#_Toc478723926)

[5.3 Standards 5](#_Toc478723927)

[5.4 Detailed description of the implementation (coding) 5](#_Toc478723928)

[6. TESTING 6](#_Toc478723929)

[7. USER GUIDE OF THE SYSTEM 7](#_Toc478723930)

[8. DISCUSSION 8](#_Toc478723931)

[9. CONCLUSION 9](#_Toc478723932)

[10. REFERENCES 10](#_Toc478723933)

[APPENDICES 11](#_Toc478723934)

[A. Instructions for installing the system 11](#_Toc478723935)

[B. Code for the system 11](#_Toc478723936)

[C. Other relevant material 11](#_Toc478723937)

# LIST OF FIGURES

[Figure 1: Gantt Chart 16](#_Toc478723910)

[Figure 2: Organization Scheme 28](#_Toc478723911)

[Figure 3: Netwrok Diagram 4](#_Toc478723912)0

[Figure 4: COCOMO FORMULA 47](#_Toc478723913)

[Figure 5: Architecture Diagram 58](#_Toc478723914)

[Figure 6: Context Level Diagram 8](#_Toc478723915)

[Figure 7: Level-0 Diagram 3](#_Toc478723916)

[Figure 8: Activity Diagram for User 3](#_Toc478723917)

[Figure 9: Activity Diagram for Crawler 3](#_Toc478723918)

[Figure 10: Sequence Diagram 3](#_Toc478723919)

[Figure 11: Use-Case Diagram 3](#_Toc478723920)

[Figure 12: Class Diagram 4](#_Toc478723921)

[Figure 13: Entity-Relation Diagram 4](#_Toc478723922)

[Figure 14: Logical Database 4](#_Toc478723923)

[Figure 15: Pyhsical Database 5](#_Toc478723924)

[Figure 16: Levenstein Algorithm 5](#_Toc478723925)

[Figure 17: Doctors Screen 5](#_Toc478723926)

[Figure 18: Home Page Screen 5](#_Toc478723927)

[Figure 19: Doctor Information Screen 5](#_Toc478723928)

[Figure 20: Testing 6](#_Toc478723929)

# LIST OF TABLE

[Table 1: 16](#_Toc478723910)

[Table 2: 28](#_Toc478723911)

[Table 3: 4](#_Toc478723912)0

[Table 4: 47](#_Toc478723913)

[Table 5: 58](#_Toc478723914)

[Table 6: 8](#_Toc478723915)

[Table 7: 3](#_Toc478723916)

[Table 8: 3](#_Toc478723917)

[Table 9: 3](#_Toc478723918)

[Table 10: 3](#_Toc478723919)

[Table 11: 3](#_Toc478723920)

[Table 12: 4](#_Toc478723921)

[Table 13: 4](#_Toc478723922)

[Table 14: 4](#_Toc478723923)

[Table 15: 5](#_Toc478723924)

[Table 16: 5](#_Toc478723925)

[Table 17: 5](#_Toc478723926)

[Table 18: 5](#_Toc478723927)

[Table 19: 5](#_Toc478723928)

[Table 20: 6](#_Toc478723929)

# 1. INTRODUCTION

The problem that this project is trying to solve is that health institutions in TRNC do not provide adequate and accurate information about the doctors in the country, so people living in TRNC have difficulties in accessing the doctors they need. With this project, users will be able to access the information they are looking for in an organized and sufficient way. The main starting point of the problem is that health institutions share information about doctors in separate places. This poses a big problem in finding the doctor needed (for example, the user may get information about the wrong doctor while searching for the doctor they need). This project aims to solve such problems.

# 2. PROJECT PLANNING AND MANAGEMENT

A.1. Preliminary Project Information

# A.1.1

|  |  |
| --- | --- |
| **Project No** | 1 |
| **Project Name** | Doctor Search System |
| **Start Date** | 01.03.2023 |
| **End Date** | 01.06.2023 |
| **Time** | 12 weeks |

# A.1.2

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Manager** | | | |
| **Name Surname** | Samet Sarıçiçek | **ID No** | 19331159 |
| **Title/Role** | Project Manager/Lead Programmer/Database Developer | | |
| **Address** | Famagusta/TRNC | | |
| **Phone** | 0542 860 15 02 | | |
| **Email** | sametsie34@gmail.com | | |

A.2 Group Information

# A.2.1

|  |  |  |  |
| --- | --- | --- | --- |
| **Student 1** | | | |
| **Name Surname** | Ayşe Dilara Öztürk | **ID No** | 19331157 |
| **Title/Role** | Interface Designer/ Tester | | |
| **Address** | Nicosia/TRNC | | |
| **Phone** | 0533 855 76 69 | | |
| **Email** | adilaraozturk@gmail.com | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Student 2** | | | |
| **Name Surname** | Kadir Kaan Sert | **ID No** | 19000096 |
| **Title/Role** | Interface Designer/Analyst | | |
| **Address** | Famagusta/TRNC | | |
| **Phone** | 0542 852 99 12 | | |
| **Email** | 19000096@emu.edu.tr | | |

# A.2.2

|  |
| --- |
| **List of Completed / Ongoing Projects of Team** |
| Samet Sarıçiçek:  CMSE405- Foster Child Graduation Project  Kadir Kaan Sert:  CMSE326- A Quality Assessment on Web Site of Girne American University  Ayşe Dilara Öztürk:  CMSE326- A Quality Assessment on Web Site of Girne American University  CMSE405- Robotic Hand Controlled by Glove using Wireless Communication |

B.1 Introduction to Project

# B.1.1

|  |
| --- |
| **Summary of Project** |
| A Doctor Search System is a type of information system that combines hardware and yazılım to make it easier for people to find and search for healthcare professionals in TRNC. This system is designed as an easy-to-use yet thorough web-based search engine where people can learn more about healthcare professionals in TRNC. |

# B.1.2

|  |
| --- |
| **Key Words** |
| HTML, CSS, SQL, JavaScript, healthcare professional, hospital, illness, disease, searching. |

# B.1.3

|  |
| --- |
| **Aim of Project** |
| One of the aims of the project is to gather a list of healthcare professionals in TRNC and make it easier for people to contact them. Our driving force of this project is to enable users living in the TRNC to learn more about healthcare professionals according to the region they live in or the country. In this way, users who want to search with this system we have developed, due to the difficulties experienced and accessing healthcare professionals and lack of data, will be able to communicate with a healthcare professional easily according to the criteria they want. For this purpose, data will be collected from hospitals all over the country, analyzed, and transferred as necessary. |

# B.1.4

|  |
| --- |
| **Innovative Aspects/Contributions of Project** |
| The primary purpose of this system is to provide a search system for patients in TRNC to reach healthcare professionals more efficiently. With this search system, users can easily access many requirements from a website, from which hospitals the healthcare professionals provide service to which cities they serve. In addition, this system is a web-based search system that collects healthcare professionals' branches, addresses, contact numbers (email addresses and phone numbers), gender, in which hospital they work (private hospital/public hospital/clinic), and whether their services are covered by insurance or not. Thanks to this system, users will be able to view the healthcare professionals they want and get information about them. In addition, with user votes on the website, the service quality of healthcare professionals will be a visible determining factor in the system. |

# B.1.5

|  |
| --- |
| **Methods to be Applied** |
| Initially, MS Project will be used to edit the completion times of the components and the project duration. Then, diagrams will be drawn using Modelio according to the system's basic purpose and the requirements for research results. Next, AdobeXD will be used for prototyping and preview purposes. Then, the front end will be made using HTML, CSS, and Javascript, and the back end will be made using NodeJS and Sequelize to make the project come to life. |

# B.1.6

|  |
| --- |
| **Economic and National Outcomes** |
| Apart from improving the overall satisfaction of the TRNC healthcare services, this system will make it easier for patients to reach doctors and will help them find the doctor they want according to the criteria they are looking for. Economically, the user will be able to search for a hospital according to his financial situation. Thus, the budget allocated for health services operations will be easily calculated. |

B.2 Reason for Starting the Project, Methods, and R&D Stages

# B.2.1

|  |
| --- |
| **1- Explain the reason for starting this project. (Max 500 characters)** |
| We did some field analyses based on health practices. As a result of these information collection processes, there is currently no doctoral information system serving a wide audience in the TRNC software market. Existing systems cannot meet the needs of users looking for a doctor, or the information they contain is not up-to-date and accurate. For these reasons, we decided that such unpopular apps should evolve, and need better improvement, increase user satisfaction, and additional functionality that people can take advantage of. |
| **2- Explain the purpose of this project.** |
| The primary purpose of developing a web-based doctor search system is to make it possible for patients to contact the healthcare professionals in the TRNC who most closely meet their individual needs at any time. Its activities make it simple for the user to locate the healthcare professionals they require quickly. Our goal is to create a system that offers benefits while also making the users’ financial choices (private hospital vs. public hospital) functionally and financially obvious in the system. Another goal is to offer users confidence in the accuracy and currentness of the system we have established. While there are some comparable systems presently on the software market, we believe that ours may advance a step further. |

|  |
| --- |
| **3- Explain**   * **output of the project** * **national/international standards if exist** * **the specific objectives of the project** * **success criteria** * **realistic constraints** |
| **Output:**  The project's outcome is a web-based application that enables residents of the TRNC to obtain healthcare professionals' information by their specific needs.  **Success Criteria:**   * Users living in the TRNC have complete access to the information of healthcare professionals.   **Realistic Constraints:**   * Inadequate and inaccurate information provided by hospital Web pages is an obstacle to practice. |
| **4- Explain**   * **the methods to be applied during R&D activities** * **applications** * **technics and tools to be used** * **standards to be followed under the workflow** |
| **Which SOFTWARE PROCESS MODEL below will you apply? Why? How? Explain.**  We decided to use The Waterfall Model to ensure that every step of the project is planned and organized. Another reason that we selected this method, we are very sure that there will be no changes in project development time. In the Waterfall Method, every step of the project is planned. If one of the steps is not completed, there is no way to advance to the next step.   1. **Feasibility and Pre-research:**   To better understand the method in which the project will be carried out, we aim to collect more information about the project by examining the systems that have used similar or the same method. To diversify the project, we aim to interview the users of the applications that have used similar or the same method and to expand the project with the answers.  **2. System Design:**  This part includes all our decisions about the system parts. Its content is the design of our system, the selection of materials and suppliers, and the determination of system parameters.   1. **Software development:**   In this part of the project, we will NodeJS to develop Scrapping and Backend modules. Also, for the front end, we will use HTML5, CSS, and React to develop the front end.     1. **Prototype implementation and testing work:**   In this step, we will create the prototype for scrapping, backend, and frontend. And we will test them parallel to save some time. For every bug or error, we will implement new fixes to improve reliability and usability.   1. **Maintenance:**   For maintenance, we will test and improve our app in short periods or with customer feedback. As result, the customer will be able to use more reliable and safe applications. |
| **5- Explain**   * **the contribution of national/international technological development if exist** * **starting new research and development projects within or outside the team** * **launch new applications or research studies in different technology areas**   **With whom can we cooperate?**  **Expectations:**  **Published work:**  **Can your output be an input for other similar national/international projects?** |
| This version of the system will be suitable for users who want to expand their search for a healthcare professional to suit their needs. The system will be designed for use by many national users collaborating with a wide range of doctors and hospitals. For this reason, the system may be of interest to health institutions that want to expand their existing systems. |

B.3 Innovative and Unique Aspects

# B.3.1

|  |
| --- |
| **1- Describe**   * **differences** * **advantages** * **superiority**   **compared to other similar projects.** |
| The fact that our system contains up-to-date and accurate information gives confidence to users. The system interface will be a user-friendly web system that can be easily used by people of all ages by preserving the site aesthetics with a simplicity that will not tire the eyes. Errors and up-to-date and inaccurate information in the system will be eliminated as quickly as possible and my system will become above the TRNC standards. Reliability and usability will be at the forefront of our system. Thanks to these features, the system will be able to continue to be used continuously. |

# B.4.1

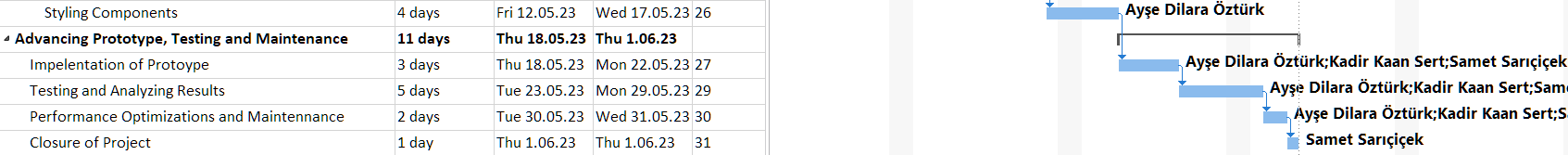
|  |
| --- |
| **2- Who can contribute to this project in your team?** |
| * Project Manager / System Analyst * Database Developer * Interface Designer * Tester |

C.1 Gantt Chart and Work Packages

# C.1.1 Gantt Chart

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu



# Figure 1: Gantt Chart

# C.1.2 List of Work Packages

|  |  |
| --- | --- |
| **Work Package No** | 1 |
| **Work Package Name** | **Project Feasibility and Pre-Research (Feasibility Analysis)** |
| **Start-End Date and Time** | 01.03.2023-17.03.2023 |
| **Related Organizations** |  |

|  |
| --- |
| **1- List the activities of work packages.** |
| **1.1 Project Process and Economic Feasibility:**   * Analysis of Similar System in Market for Need of the Project * Project Requirements Analysis * Collecting Information from Stakeholders * Estimating Budget and Worth of Project * Work Distribution   **1.2 Technological Feasibility:**   * Hardware and Software Requirements Analysis * Determining the Resources that will be used for this Project. * Testing the Resources and Deciding How to Implement * Generating Requirements |
| **2- Describe the methods and parameters that will be used for the work package.** |
| In the initiation of the project, we will analyze and compare similar projects on the market. This will give us an insight into the scope and aim of the project. Then we will research and analyze the resources we will use for this specific project. This part is critical because the system will be built on top of resources. |
| **3- List the experiments, tests, and analyses in the work package.** |
| * Economic Feasibility Analysis * Technical Feasibility Analysis * Resource Analysis and Testing |
| **4- List the output of the work package and its success criteria.** |
| **Outputs:**   * Creation of task distribution. * Research similar projects and systems. * Decision to progress or leave the project. * Cost estimation.   **Success Criteria:**   * To familiarize team members with the project. * To have an idea about the system overview and concepts. * To better understand the requirements. * To understand the possible risks that may occur in the system. |
| **5- Explain the relation of output with other work packages** |
| This work package is the basic step to perform other work packages as the others will be developed according to this one. It will give an estimation of whether the project will succeed if certain steps are taken. |

|  |  |
| --- | --- |
| **Work Package No** | 2 |
| **Work Package Name** | **Analysis and Design** |
| **Start-End Date and Time** | 18.03.2023-04.04.2023 |
| **Related Organizations** | - |

|  |
| --- |
| **1- List the activities of work packages.** |
| * Making job description * Determination of system requirements * Determining the resources required for the system (machines, tools, computers, etc.) * Follow-up of team members and job descriptions * Drawing the necessary diagrams to make the application more understandable. |
| **2- Describe the methods and parameters that will be used for the work package.** |
| * Division of work among team members * Modeling the project and drawing the necessary diagrams. |
| **3- List the experiments, tests, and analysis in the work package.** |
| * Drawing schematics in accordance with the system model * Compilation of information collected from team members who distribute work. * Create samples. * Getting feedback from people who know about the subject |
| **4- List the output of the work package and its success criteria.** |
| **Outputs:**   * Compliant requirements report. * Some design examples of the system * Detailed models showing the overall flow of the system. * Controlling the distribution of work   **Success Criteria:**   * Understanding the system structure * To find solutions to the possible risks of the system * Get feedback from specific users. * Making market analysis |
| **5- Explain the relation of output with other work packages** |
| This system will use the requirements analysis package as an input to perform diagramming for requested business operations, division of labor among team members, further analysis, and risk management practices. The output of this phase will be used as input in the coding phase (Development of System Software). |

|  |  |
| --- | --- |
| **Work Package No** | 3 |
| **Work Package Name** | **Development of System Software** |
| **Start-End Date and Time** | 05.04.2023-17.05.2023 |
| **Related Organizations** | - |

|  |
| --- |
| **1- List the activities of work packages.** |
| * Easy to understand interface. * System integration of sites and databases from which data will be obtained. * Develop Database to store data. * Programming the system * Application of requirements to the system |
| **2- Describe the methods and parameters that will be used for the work package.** |
| * MySQL for database (data storage) * HTML to create the main structure of the system. * CSS to style the system. * JavaScript to make the system functional and more dynamic. * NodeJS to connect the system to the database. |
| **3- List the experiments, tests and analysis in the work package.** |
| * Checking compliance of requirements * Checking the data to be used * Checking the functionality of the system * Develop enough storage space. * Checking the functionality of the interface |
| **4- List the output of work package and its success criterias.** |
| **Outputs:**   * Ready and functional system example * Can be used to test the system.   **Success Criteria:**   * Sufficient database space * Checking the information and making sure that it is reliable. * Well coded system * User friendly interface |
| **5- Explain the relation of output with other work packages** |
| At the end of this workpack we will have a functional prototype of the system and the functional interface of the system. After these work packages, the system will be ready to test for bugs. In addition, it will be decided whether the interface is user-friendly or not. This phase will be used as input directly to the Testing team. |

|  |  |
| --- | --- |
| **Work Package No** | 4 |
| **Work Package Name** | **Advancing Prototype, Testing and Maintenance** |
| **Start-End Date and Time** | 18.05.2023-01.06.2023 |
| **Related Organizations** | - |

|  |
| --- |
| **1- List the activities of work packages.** |
| * Security test * Cookies Checking Testing * Path test * Performance test * Safety Test * Speed Test |
| **2- Describe the methods and parameters that will be used for the work package.** |
| * Black box and white box penetration testing for security * User acceptance testing for requirements. * Model-based tests * Unit testing * Speed test for connection speed * General testing practices for performance testing |
| **3- List the experiments, tests and analysis in the work package.** |
| * The speed test will check how fast the system processes data. * Performance test will examine the system. * Black box and white box testing will simulate an external hacking attack. * Model-based tests show whether the system performs the desired behavior. * Unit testing will check the design and behavior of applications and make sure they work as intended. |
| **4- List the output of work package and its success criterias.** |
| **Outputs:**   * Results and reports of tests * Bug reports * Suggestions for solutions for problems   **Success Criteria:**   * The system and its applications work and work as desired and are ready to use. * Fixing the errors in the tests. * Making the system faster. |
| **5- Explain the relation of output with other work packages** |
| If this work package is completed completely and without any problems, it means that the system is working perfectly, there are no errors or problems, and it is ready to use. This package tests every coding step. According to the results of the test results, the program will be arranged, and the problems will be corrected. This cycle will continue until it runs without errors. Finally, it will be ready to use without any errors. |

# C.1.3 List of Milestones

|  |  |  |
| --- | --- | --- |
|  | **Description of Output** | **Expected Time Interval** |
| 1 | **Project Feasibility and Pre-Research (Feasibility Analysis)** | 01.03.2023-15.03.2023 |
| 2 | **System Analysis and Design** | 16.03.2024-24.03.2023 |
| 3 | **System Development** | 25.03.2023-19.04.2023 |
| 4 | **Advancing Prototype, Testing and Maintenance** | 20.04.2023-04.05.2023 |

## Table 1: List of Milestones

# C.1.4 List of Risks (see following example, find other risks of your Project!)

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Probability | Effects | Your Strategy |
| The time required to develop the software is underestimated. | High | Serious | Time problems can be solved in many ways. Such as fair job distribution, cooperation, well-planned team management and focusing on essential features. |
| Software tools cannot work together in an integrated way. | High | Tolerable | We will include the tools integration in our research to not face with such a problem |
| Customers fail to understand the impact of requirements changes. | Moderate | Tolerable | Because odSDLC Model that we have selected we should get feed-back from customer regularly |
| Critical bugs in algorithm | Moderate | Serious | RReview of the project algorithm and constantly test to find what causes the bug |
| Any requirement errors on the device where the program will be executed. | Modareta | Tolerable | Review of the requirements and rechecking connections between execution and requirements. |
| Code generated by code generation tools is inefficient. | Moderate | Insignificant | Variables, Class Names and Functions can be written manually |
| Hosting and Server Crash | Low | Serious | Taking back-up regularly and choosing correct hosting package & provider for the project. |
| The database used in the system cannot process as many transactions per second as expected. | Moderate | Serious | Choosing High-Performance Databse. |

## Table 2: List of Risks

C.2 Project Management and Organization

# C.2.1 Project Team

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Personnel Name** | **Title** | **ID** | **Education Status** | **Graduation Date** | **Date of Starting Work** | **Idea Owner** |
| Samet Sarıçiçek | Project Manager/Lead Programmer/Database Developer | 19331159 | Undergraduate | 2024 | 01.03.2023 | YES |
| Ayşe Dİlara Öztürk | Interface Designer/ Tester | 19331157 | Undergraduate | 2024 | 01.03.2023 | YES |
| Kadir Kaan Sert | Interface Designer/Analyst | 19000096 | Undergraduate | 2024 | 01.03.2023 | YES |

## Table 3: Project Team

# C.2.2 Organization Scheme

metin, diyagram, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Figure 2: Organization Scheme

D.1 Economic Forecasts

|  |
| --- |
| **1- Evaluate the commercialization potential of project outcomes. List possible risks here.** |
| Many industries today use accepted financial practices to guide their decisions to commercialize new technology. We will use these applications to sustain our project financially. While presenting this project to the software market, our first goal is to ensure that all age groups in the TRNC have access to doctor information and choose the doctor who meets the criteria they want. Since the main purpose of the project is not to provide financial accumulation, financial expectations are quite low. However, this project can be commercialized, and financial benefits can be obtained by contracting with private companies. The risks that may arise from the commercialization of this project may be that the people in need must access the doctor's information for a certain fee and cannot receive the health services they want. |

|  |  |
| --- | --- |
| **2- List your expectations to your team which is come by your project** | |
| Time-to-market (month): | June 2023 |
| The expected increase in sales revenue (%): | Unknown |
| The expected increase in market share (%): | Unknown |
| Time to start to gain: | Unknown |

D.2 National Outcomes

|  |
| --- |
| **1- Specify the output that may be subject to patent, utility model and industrial design registration in the project.** |
| This project can be adapted for each person. Thanks to system architecture we can afford and handle incoming requests and problems for each different situation.  Although there are already similar systems on the market, we will implement some additional functionality that may be the subject of patents. Since this is a software project, it does not apply to our utility model and industrial design registration system. Any collaboration with other companies that requires these issues will be handled by that company. |
| **2- Explain the potential of the project and its outputs that may have an effect on social life, education, health etc.** |
| This project will bring many conveniences to the health sector as the user can choose the doctor who meets his criteria wherever he wants. By entering the user criteria, he will be able to find the doctor he wants quickly, saving time. It will be able to quickly access the information of the doctors available in the system. In addition, by keeping our system up to date, users will be able to gain their trust easily. |
| **3- Explain the positive and negative effects of project outputs on the environment and human beings.** |
| The system we have developed receives information from the Ministry of Health. Since our system will be reliable, it will create positive results for users. Since no financial income is intended from the system, it will be available to everyone. The system has no negative effects on users. |

(M013) Instrument / Equipment / Software / RELEASE PURCHASES

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name** | | **Doctor Search System** | | | | | | | | | |
| **Line no** | **Instrument / Equipment / Software / Publication Name** | | **No. of Item** | **Capacity** | **Technical specification** | **Purpose of Project Activities** | **Post-Project Place of Use / Purpose** | | **Unit Price (USD)** | **Unit Price (TL)** | **Total Amount (TL)** |
| **R & D** | **Production** |
| **1** | **Laptop** | | **3** |  | **512gb minimum with 100gb unused space and Core i7 quadcore 2.2 GHz or faster** | **Involved in every task of the project** | **YES** | **YES** | **110** | **13.000** | **39.000** |
| **2** | **Internet Connection** | | **1** |  | **Min 20Mbps** | **Connection** | **YES** | **YES** | **190** | **1.200** | **1.200** |
| **3** | **MS Project** | | **1** |  | **Project**  **Management**  **Software** | **Planning** | **YES** |  | **30** | **200** | **200** |
| **4** | **Office 365** | | **1** |  | **Collection of**  **office-related**  **applications** | **Management** | **YES** |  | **100** | **660** | **660** |
| **5** | **Modelio** | | **1** |  | **Tools for drawing diagrams** | **UML Diagrams** | **YES** |  | **-** | **-** | **-** |
| **6** | **MySQL** | | **1** |  | **Tool for creating database** | **Database**  **Operations** |  | **YES** |  | **5.000** | **3.300** |
| **7** | **Php** | | **1** |  | **Language for**  **developing the web application** | **Coding Purpose** |  | **YES** |  | **579** | **2.024** |
| **8** | **JavaScript** | | **1** |  | **Language for**  **developing the web application** | **Coding Purpose** |  | **YES** |  | **309** | **3.790** |
| **9** | **HTML** | | **1** |  | **Creating electronic**  **documents** | **Coding Purpose** |  | **YES** |  | **-** | **-** |
|  |  | |  |  |  |  |  |  |  | **TOTAL** | **50.174** |

## Table 4: Expenses

(M030) Quarterly Estimated Cost Form (TL)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Project Name: Doctor Search System** | | | | |
| **Cost Item** | **2022-2023** | | **TOTAL**  **(TL)** | **TOTAL COST RATE OF CONTENTS (%)** |
| **I** | **II** |
| **Personnel** | 8000 |  | 8000TL | 26.35% |
| **Travel** | 2700 | - | 2700TL | 8.89% |
| **Instrument / Equipment / Software / Publications** | 19525 | - | 19525TL | 64.31% |
| **Domestic Works Made by R & D and Testing Institutions** | - | - | - | - |
| **International Works Made by R & D and Testing Institutions** | - | - | - | - |
| **Domestic Services Procurement** | - | - |  |  |
| **Overseas Service Procurement** | 135 | - | 135TL | 0.44%0. |
| **Material** | Based on personal needs | - | - | - |
| **TOTAL COST** | 30360TL | - | 30360TL | 100 |
| **CUMULATIVE COST** | 30360TL | - | 30360TL | 100 |
| **IN THE PROJECT TOTAL MAN-MONTH** | | | 30360TL | |

## Table 5: Estimated Cost Form

APPENDIX

**Activity Table**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activity** | **Predicates** | **Duration(day)** | **Optimistic Time(day)** | **Most Likely(day)** | **Pessimistic(day)** | **Variance** | **Standard Deviation** | **Probability of Sucessful Completion Rate of Each Path (%)** | **Expected Time(day)** |
| A | - | 1 | 0.5 | 1 | 2 | 0.0625 | 0.5 | 90 | 1.083 |
| B | A | 1 | 0.5 | 1 | 1.5 | 0.027556 | 0.40743097574926725249 | 95 | 1 |
| C | A | 1 | 0.5 | 1 | 1.5 | 0.027556 | 0.40743097574926725249 | 92 | 1 |
| D | B, C | 1 | 1 | 1 | 2 | 0.027556 | 0.40743097574926725249 | 93 | 1.116 |
| E | D | 3 | 2 | 3 | 4 | 0.110889 | 0.57706152185014034256 | 98 | 3 |
| F | E | 2 | 1 | 2 | 3 | 0.110889 | 0.57706152185014034256 | 97 | 2 |
| G | F | 3 | 2 | 3 | 5 | 0.25 | 0.7071067811865475244 | 96 | 3.166 |
| H | G | 2 | 1 | 2 | 4 | 0.25 | 0.7071067811865475244 | 96 | 2.166 |
| I | H | 1 | 0.5 | 1 | 2 | 0.0625 | 0.5 | 95 | 1.083 |
| J | I | 3 | 1 | 3 | 4.5 | 0.339889 | 0.76354436675284298767 | 89 | 2.916 |
| K | J | 4 | 2 | 4 | 5 | 0.25 | 0.7071067811865475244 | 91 | 3.833 |
| L | K, J | 2 | 1 | 2 | 3 | 0.110889 | 0.57706152185014034256 | 84 | 2 |
| M | L | 5 | 3 | 5 | 7 | 0.443556 | 0.8160882305241266007 | 82 | 5 |
| N | M | 6 | 4 | 6 | 7 | 0.25 | 0.7071067811865475244 | 87 | 5,833 |
| O | N | 3 | 1 | 3 | 5 | 0.443556 | 0.8160882305241266007 | 89 | 4 |
| P | O, M | 2 | 1 | 2 | 6 | 0.693889 | 0.91268833672837082004 | 87 | 2,500 |
| Q | P | 3 | 2 | 3 | 5 | 0.25 | 0.7071067811865475244 | 83 | 3,166 |
| R | Q | 7 | 6 | 7 | 10 | 0.443556 | 0.8160882305241266007 | 89 | 7,333 |
| S | P | 2 | 1.5 | 2 | 4 | 0.173056 | 0.64498061986388397219 | 92 | 2,250 |
| T | S | 4 | 3.5 | 4 | 5 | 0.0625 | 0.5 | 87 | 4,083 |
| U | T | 4 | 3 | 4 | 5 | 0.110889 | 0.57706152185014034256 | 84 | 4 |
| V | U | 3 | 2 | 3 | 4 | 0.110889 | 0.57706152185014034256 | 89 | 3 |
| W | V | 5 | 3 | 5 | 7 | 0.443556 | 0.8160882305241266007 | 96 | 5 |
| X | W | 2 | 1 | 2 | 4 | 0.25 | 0.7071067811865475244 | 95 | 2,166 |
| Y | X | 1 | 1 | 1 | 3 | 0.110889 | 0.57706152185014034256 | 96 | 1,333 |

## Table 6: Activity Analysis

**Path Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Paths** | **Total Expected Time for Each Path** | **Variance of Each Path** | **Standard Deviation of Each Path** |
| 1 | AD | 2.199 | 0.090056 | 0.30009331881932993145 |
| 2 | ABEFGHIJKLMNOPQRSTUVWXY | 71.911 | 5.361448 | 2.3154800798106642317 |
| 3 | ABEFGHIJKLMPQRSTUVWXY | 62.078 | 4.667892 | 2.1605304904120191564 |
| 4 | ABEFGHIJKLMNOPQRSTUVWXY | 71.911 | 5.361448 | 2.3154800798106642317 |
| 5 | ABEFGHIJLMPQRSTUVWXY | 58.245 | 4.417892 | 2.1018782076990093766 |
| 6 | ACEFGHIJKLMNOPQRSTUVWXY | 71.911 | 5.361448 | 2.3154800798106642317 |
| 7 | ACEFGHIJKLMQRSTUVWXY | 59.578, | 3.974003 | 1.9934901554810848877 |
| 8 | ACEFGHIJLMNOPQRSTUVWXY | 68.078 | 5.111448 | 2.2608511671492221427 |
| 9 | ACEFGHIJLMPQRSTUVWXY | 58.245 | 4.417892 | 2.1018782076990093766 |

## Table 7: Critical Path Analysis

**Network Diagram (Indicating the Critical Path with Red Lines)**

**daire, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu**

## Figure 3: Network Diagram

**CPM (Critical Path Management) analysis by using PERT (defining paths)**

The first one with bold font is our critical path having days. The calculation of the first path is shown, other paths are calculated in a similar way by checking the table values in PERT Calculation Section.

**PERT Calculation**

Expected time calculated using expected duration values. Variance calculated by variance values of each process. Standard deviation calculated by taking square root of variance. You can also see all other possible paths. The longest path “**ABEFGHIJKLMNOPQRSTUVWXY** “indicated in bold color which has 71.911 days is our critical path.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activity** | **Predicates** | **Duration(day)** | **Optimistic Time(day)** | **Most Likely(day)** | **Pessimistic(day)** | **Variance** | **Standard Deviation** | **Expected Time(day)** |
| A | - | 1 | 0.5 | 1 | 2 | 0.0625 | 0.5 | 1.083 |
| B | A | 1 | 0.5 | 1 | 1.5 | 0.027556 | 0.40743097574926725249 | 1 |
| C | A | 1 | 0.5 | 1 | 1.5 | 0.027556 | 0.40743097574926725249 | 1 |
| D | B, C | 1 | 1 | 1 | 2 | 0.027556 | 0.40743097574926725249 | 1.116 |
| E | D | 3 | 2 | 3 | 4 | 0.110889 | 0.57706152185014034256 | 3 |
| F | E | 2 | 1 | 2 | 3 | 0.110889 | 0.57706152185014034256 | 2 |
| G | F | 3 | 2 | 3 | 5 | 0.25 | 0.7071067811865475244 | 3.166 |
| H | G | 2 | 1 | 2 | 4 | 0.25 | 0.7071067811865475244 | 2.166 |
| I | H | 1 | 0.5 | 1 | 2 | 0.0625 | 0.5 | 1.083 |
| J | I | 3 | 1 | 3 | 4.5 | 0.339889 | 0.76354436675284298767 | 2.916 |
| K | J | 4 | 2 | 4 | 5 | 0.25 | 0.7071067811865475244 | 3.833 |
| L | K, J | 2 | 1 | 2 | 3 | 0.110889 | 0.57706152185014034256 | 2 |
| M | L | 5 | 3 | 5 | 7 | 0.443556 | 0.8160882305241266007 | 5 |
| N | M | 6 | 4 | 6 | 7 | 0.25 | 0.7071067811865475244 | 5,833 |
| O | N | 3 | 1 | 3 | 5 | 0.443556 | 0.8160882305241266007 | 4 |
| P | O, M | 2 | 1 | 2 | 6 | 0.693889 | 0.91268833672837082004 | 2,500 |
| Q | P | 3 | 2 | 3 | 5 | 0.25 | 0.7071067811865475244 | 3,166 |
| R | Q | 7 | 6 | 7 | 10 | 0.443556 | 0.8160882305241266007 | 7,333 |
| S | P | 2 | 1.5 | 2 | 4 | 0.173056 | 0.64498061986388397219 | 2,250 |
| T | S | 4 | 3.5 | 4 | 5 | 0.0625 | 0.5 | 4,083 |
| U | T | 4 | 3 | 4 | 5 | 0.110889 | 0.57706152185014034256 | 4 |
| V | U | 3 | 2 | 3 | 4 | 0.110889 | 0.57706152185014034256 | 3 |
| W | V | 5 | 3 | 5 | 7 | 0.443556 | 0.8160882305241266007 | 5 |
| X | W | 2 | 1 | 2 | 4 | 0.25 | 0.7071067811865475244 | 2,166 |
| Y | X | 1 | 1 | 1 | 3 | 0.110889 | 0.57706152185014034256 | 1,333 |

## Table 8: Pert Analysis

* **COCOMO ESTIMATION**
* **DI**

|  |  |
| --- | --- |
| Factor | Complexity Value |
| Data Communication | 3 |
| Distributed Data Processing | 4 |
| Performance Criteria | 5 |
| Heavily Utilized Hardware | 2 |
| High Transaction Rates | 4 |
| Online Data Entry | 5 |
| Online Updating | 5 |
| End-user Efficiency | 5 |
| Complex Computations | 1 |
| Reusability | 3 |
| Ease of Installation | 3 |
| Ease of Operation | 3 |
| Portability | 1 |
| Maintainability | 5 |
| **DI=** | **47** |

## Table 9: DI Table

* **UFP**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Functions** | **Simple No** | **Simple Weight** | **Average No** | **Average Weight** | **Complex No** | **Complex Weight** | **Total** |
| **User Input** | 4 | 3 | 5 | 4 | 1 | 6 | 4\*3+5\*4+1\*6 = 38 |
| **User Output** | 4 | 4 | **5** | 5 | 6 | 7 | 4\*4+5\*5+6\*7 = 83 |
| **User Query** | 2 | 3 | 3 | 4 | 5 | 6 | 2\*3+3\*4+5\*6 = 48 |
| **Internal Files** | 4 | 7 | 5 | 10 | 5 | 15 | 4\*7+5\*10+5\*15 = 153 |
| **External Interfaces** | 0 | 5 | 0 | 7 | **0** | 10 | 0\*5+0\*7+0\*10 = 0 |
| **Count Total** |  |  |  |  |  | = | **322** |

## Table 10: UFP Table

* **FP**

FP = UFP\*[0.65 + 0.01\*DI]

FP = 322 \* [0.65 + 0.01\*47]

FP = 360.64

\*KLOC calculated using High-Level Language (Javascript) = 47

KLOC = (360.64\*47) /1000.

KLOC = 16.95

\*Our Project is basic so we will use Basic COCOMO Table, and our Project is Semidetached

masa içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Figure 4: COCOMO Formula

E = 3.0\*(16.95)1.12 = 71.41

D = 2.5(71.41)0.35= 11.13

SS = 71.41/11.13 = 6.41

P = 16.95/71.41 = 0.23

**3-Calculating probability of successful completion rate for each path**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Paths** | **Total Expected Time for Each Path** | **Specified Time** | **Variance of Each Path** | **Standard Deviation of Each Path** | **Probability(Z-Score)** |
| 1 | AD | 2.199 | 100 | 0.090056 | 0.300093318 | 326.038 |
| 2 | ABEFGHIJKLMNOPQRSTUVWXY | 71.911 | 100 | 5.361448 | 2.31548007 | 12.185 |
| 3 | ABEFGHIJKLMPQRSTUVWXY | 62.078 | 100 | 4.667892 | 2.16053049 | 18.434 |
| 4 | ABEFGHIJKLMNOPQRSTUVWXY | 71.911 | 100 | 5.361448 | 2.31548007 | 12.185 |
| 5 | ABEFGHIJLMPQRSTUVWXY | 58.245 | 100 | 4.417892 | 2.10187820 | 20.000 |
| 6 | ACEFGHIJKLMNOPQRSTUVWXY | 71.911 | 100 | 5.361448 | 2.31548007 | 12.185 |
| 7 | ACEFGHIJKLMQRSTUVWXY | 59.578, | 100 | 3.974003 | 1.9934901554810848877 | 20.718 |
| 8 | ACEFGHIJLMNOPQRSTUVWXY | 68.078 | 100 | 5.111448 | 2.2608511671492221427 | 14.520 |
| 9 | ACEFGHIJLMPQRSTUVWXY | 58.245 | 100 | 4.417892 | 2.1018782076990093766 | 20.000 |

**4-Crashing Approach**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activity** | **Normal Time (day)** | **Crash Time (day)** | **Normal Cost ($)** | **Crash Cost ($)** | **Reduce / Crash Cost Per Day ($)** |
| A | 1 | 1 | 20 | 35 | 15 |
| B | 1 | 1 | 25 | 30 | 5 |
| C | 1 | 1 | 30 | 40 | 10 |
| D | 1 | 1 | 20 | 25 | 5 |
| E | 3 | 2 | 35 | 40 | 5 |
| F | 2 | 1 | 35 | 45 | 10 |
| G | 3 | 2 | 40 | 50 | 10 |
| H | 2 | 1 | 30 | 35 | 5 |
| I | 1 | 1 | 40 | 45 | 5 |
| J | 3 | 1 | 65 | 80 | 15 |
| K | 4 | 2 | 35 | 50 | 15 |
| L | 2 | 1 | 30 | 40 | 10 |
| M | 5 | 3 | 30 | 35 | 5 |
| N | 6 | 4 | 50 | 62 | 12 |
| O | 3 | 1 | 45 | 53 | 8 |
| P | 2 | 1 | 45 | 60 | 15 |
| Q | 3 | 1 | 55 | 67 | 12 |
| R | 7 | 4 | 75 | 85 | 10 |
| S | 2 | 1 | 60 | 65 | 5 |
| T | 4 | 2 | 55 | 70 | 15 |
| U | 4 | 2 | 25 | 30 | 5 |
| V | 3 | 1 | 20 | 35 | 15 |
| W | 5 | 2 | 15 | 20 | 5 |
| X | 2 | 1 | 20 | 30 | 10 |
| Y | 1 | 1 | 20 | 25 | 5 |

Total Cost = 720$ day

Weekly = 720 \* 7 = 5040$

We chose B, C, D, E, G, H, M, O, S to crash (reduce).

Calculated cost after crashing: 720 + 5 + 10 + 5 + 5 + 5+ 5 + 8 = 763 $/day and 5045 $/week.

Thus, with an extra 43 dollars every day, we can recuperate from a one-week slump (43 x 7 = 301 $/week).

# 3. REQUIREMENTS ANALYSIS

## 3.1 Functional Requirements

## Scrap Data

*Description and Priority*

This feature shows the scraping and processing of data. Using predefined sources, data is retrieved from specified sources and saved in the system database. This ensures a regular and diverse flow of data into the system.

*Stimulus/Response Sequences*

This feature works by activating the module at time intervals determined by the system. The system sets a time of the day for this module to work, and the same module works in parallel during that time and transfers data to the system.

*Functional Requirements*

FREQ-1: Crawler shall scrap data in predetermined time intervals.

* Specified time interval must be format of hour.

## Organize Data

*Description and Priority*

This feature shows the organization and clearing of scrapped data. The data scrapped is not suitable for saving to the database, this feature will format data to most suitable form.

*Stimulus/Response Sequences*

This feature works after the scrapping process and before saving to the database.

*Functional Requirements*

FREQ-1: Crawler shall organize scraped data.

* Data should be formatted in a way that can be saved to the databse.

## Search Doctor

*Description and Priority*

The user will be able to search user with any given data. This feature has highest priority because this is one of the main functions of the system.

*Stimulus/Response Sequences*

User should write anything he/she wants to and after clickingthe search button system will be given a list of doctors which related to data user provided. And after that user should click the list item and view the doctor info.

*Functional Requirements*

FREQ-1: User should be able to search Doctors.

* + - * + Any input will result as list of the doctors.

## Filter Doctor

*Description and Priority*

The user will be able to filter doctors according to speciality, insurances, titles, rating. This feture has high priority because this is one of the main functions.

*Stimulus/Response Sequences*

User should select filter options he/she wants to filter and after clicking the filter button filtered doctor list should appear on the screen.

*Functional Requirements*

FREQ-1: User should be able to filter Doctors.

* + Users can filter by location, specialty, hospital type, rating, price, title.

FREQ-2: User should paginate between list of doctors.

The user can will know which page he/she at it.

## Sort Doctor

*Description and Priority*

The user will be able to sort listed doctors. User can sort the doctors based on name, rating, and specialty.

*Stimulus/Response Sequences*

Users should select sort option of the sort doctors.

*Functional Requirements*

FREQ-1: User should be able to sort doctors.

* + Users can sort by name, rating, and specialty.

## View and Inspect Doctor

*Description and Priority*

The user will be able to view doctors. User can get information about the doctor as name, speacilaity, rating, insurances doctor cover, hospital current working in and location.

*Stimulus/Response Sequences*

Users should click the doctor from the main list or from the search dropdown.

*Functional Requirements*

FREQ-7: User should be able to view Doctor Information page.

Page should include Doctor name, surname, phone, email, specialty, rating, currently working hospital, location, and map.

## Manage Doctors

*Description and Priority*

This featureonly available for the admin. Admin can and should be able to manage doctoral information through a database. The doctor can update, delete, insert, and view doctor information through the DBMS.

*Stimulus/Response Sequences*

Admin should open the MySQL workbench and select related database.

*Functional Requirements*

FREQ-8: Admin shall be able to add Doctor.

* Admin should add the doctor into the system with detailed information.

FREQ-9: Admin shall be able to delete doctor.

* Admin can delete doctor, but system should store the deleted data for a while.

FREQ-10: Admin should be able to update Doctor.

FREQ-11: Admin should view the Doctor info.

## 3.2 Non-Functional Requirements

## Performance Requirements

Caching and compression will be utilized in requests to enhance performance. With caching, the user can access the same data without having to wait for subsequent requests because it is stored in a temporary memory. The requested data will be supplied in a compressed and condensed form utilizing compression methods. The system's performance will significantly improve as a result.

## Safety Requirements

All the data in the system will be backed up regularly by the admin.

## Security Requirements

Request headers that defend against Cross-Site and MIME type attacks will be utilized to prevent the Cross-Site Scripting (XSS) attack.

*Attributes:*

* Request headers should involve X-XSS protection header.
* Requests rate should be limited to 50 requests per minute.
* CORS should be utilized to prevent Cross-origin resource sharing.

## Scalability Requirements

System written in OOP and implements SOLID principles. If any module wants to be added later, system scalability will be preserved.

## 3.3 Realistic constraints

## Health institutions in the TRNC do not provide enough information, making it impossible to provide any information about doctors. And since it is a legal offense to obtain and accumulate any data on the internet without the permission of these institutions, we only accumulate and process the information available on the internet and transfer it to the user. To obtain more information, we should serve as a health institution service, but it is impossible under these conditions. Since the information provided is already shared by health institutions, there is no such thing as using doctors' information for personal purposes. Thus, the doctor does not need to have any ethical concerns whatsoever. In addition, since this source of information will not be sufficient in the long run, it is inevitable that this doctor's information will have to be accessed in another way.

## 3.4 Ethical issues

# There is no information that can be used to commit a crime or to prevent a crime from being uncovered. No one can commit an unethical act using this system4. DESIGN

## 4.1 High level design (architectural)

metin, diyagram, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Figure 5: Architecture Diagram

## 4.2 Software design

4.2.1 Context Level Diagram

***diyagram, metin, taslak, daire içeren bir resim

Açıklama otomatik olarak oluşturuldu***

# Figure 6: Context Level Diagram

4.2.3 Level-0 Diagram

metin, diyagram, plan, teknik çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Figure 7: Level-0 Diagram

4.2.4 Activity Diagram for User2.4 Activity Diagram for User

***metin, diyagram, ekran görüntüsü, taslak içeren bir resim

Açıklama otomatik olarak oluşturuldu*** Figure 8: Activity Diagram for User

4.2.5 Activity Diagram for Crawler

***metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu***

# Figure 9: Activity Diagram for Crawler

4.2.6 Sequence Diagram

***metin, diyagram, çizgi, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu***

# Figure 10: Sequence Diagram

diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu4.2.7 Use Case Diagram

# Figure 11: Use-Case Diagram

**Use Case Glossary**

|  |  |
| --- | --- |
| **Use Case - 1** | Add Doctor |
| **Actor** | Admin |
| **Goal** | To create a doctor information |
| **Preconditions** | doctor must be living in Cyprus |
| **Stimulus** | Admin wants to create a new doctor account |
| **Scenario** | 1) Admin collects doctor information  2)Admin uploads doctor information to the system  3) Admin can update existing doctor information |
| **Exceptions** | Doctor information already available |

|  |  |
| --- | --- |
| **Use Case - 2** | Delete Doctor |
| **Actor** | Admin |
| **Goal** | To delete a doctor information |
| **Preconditions** | Doctor information already available |
| **Stimulus** | Admin wants to remove doctor from system |
| **Scenario** | 1- The admin should access the doctor information available in the system.  2- Admin should remove the doctor information from the system. |
| **Exceptions** | Doctor not available in the system |

|  |  |
| --- | --- |
| **Use Case - 3** | Edit Doctor |
| **Actor** | Admin |
| **Goal** | To edit a doctor information |
| **Preconditions** | Doctor must be present in the system |
| **Stimulus** | Admin wants to edit doctor information. |
| **Scenario** | 1-Admin should go to the doctor's information page.  2-Admin should update the doctor's information or complete the missing information. |
| **Exceptions** | The doctor's information is not missing. |

|  |  |
| --- | --- |
| **Use Case - 4** | View Doctor |
| **Actor** | Admin |
| **Goal** | To view doctor information |
| **Preconditions** | Doctor must be present in the system |
| **Stimulus** | The admin may have wanted to check the doctor's information. |
| **Scenario** | 1-Admin must enter the application  2-You should view the doctor's information |
| **Exceptions** | Doctor does not present in the system |

|  |  |
| --- | --- |
| **Use Case - 5** | Edit UI |
| **Actor** | Admin |
| **Goal** | To change the system UI |
| **Preconditions** | Doctor must be present in system |
| **Stimulus** | The system needs a change |
| **Scenario** | 1-Admin must login to the system  2-Admin should design a new UI  3-Admin should integrate the designed UI into the system |
| **Exceptions** | Displayed error in new UI |

|  |  |
| --- | --- |
| **Use Case - 6** | Update Doctor |
| **Actor** | Admin |
| **Goal** | Admin, Update doctor information |
| **Preconditions** | The system displays the register page |
| **Stimulus** | Doctors' information page is needed to update |
| **Scenario** | 1-Admin must login to the system  2-Recognizes that the doctor's page needs updating  3-Updates doctor information in the system |
| **Exceptions** | Doctors page is not valid |

**For User**

|  |  |
| --- | --- |
| **Use Case - 1** | Filter Doctor |
| **Actor** | User |
| **Goal** | The user should filter the characteristics of the doctor they are looking for |
| **Preconditions** | User must have internet access |
| **Stimulus** | The user wishes to view doctor information |
| **Scenario** | 1-The user must access the internet  2-Go to the application page  3-The user must enter the doctor's criteria into the system  4-The user should click on the doctor's page and view the information. |
| **Exceptions** | User entered doctor criteria incorrectly |

|  |  |
| --- | --- |
| **Use Case - 2** | Seacrh Doctor |
| **Actor** | User |
| **Goal** | The user must enter the name of the doctor they are looking for and search in the system. |
| **Preconditions** | The user must know the name of the doctor they are looking for |
| **Stimulus** | The user should specifically seek out a doctor |
| **Scenario** | 1-The user must access the internet  2-Go to the application page  3-The user must enter the doctor's name into the system  4-The user should click on the doctor's page and view the information. |
| **Exceptions** | The user entered the doctor's name incorrectly |

|  |  |
| --- | --- |
| **Use Case - 3** | Sort Doctor |
| **Actor** | User |
| **Goal** | The user doctors sort by score |
| **Preconditions** | The system displays the register page |
| **Stimulus** | The user must have internet access |
| **Scenario** | 1-The user must access the internet and enter the system  2-Enter the user criteria and filter the doctors  3-The system will rank the doctors according to their scores |
| **Exceptions** | The system may rank the doctors wrong |

|  |  |
| --- | --- |
| **Use Case - 4** | View and Inspect Doctor |
| **Actor** | User |
| **Goal** | The user should want to review the information page of the doctor they are looking for |
| **Preconditions** | The user must have found the doctor they are looking for |
| **Stimulus** | The user should want to review the information of the doctor they are looking for |
| **Scenario** | 1-The user must enter the system with internet access  2-The user should either search the system with the name of the doctor or find the doctor he is looking for by filtering his criteria.  3-The user can review the doctor's information by clicking on the information page. |
| **Exceptions** | The user entered the doctor incorrectly or entered the criteria incorrectly |

|  |  |
| --- | --- |
| **Use Case - 5** | View Info About Website |
| **Actor** | User |
| **Goal** | Getting the user information about the system |
| **Preconditions** | User must login to the system |
| **Stimulus** | The user wants to learn something about website |
| **Scenario** | 1-The user must log in to the system with internet access  2- Click on the "information about website" button  3-After clicking the button, you will be directed to the page where you can get information about the system. |
| **Exceptions** | There may be a problem with the user's internet access |

4.2.8 Class Diagram

metin, ekran görüntüsü, diyagram, dikdörtgen içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Figure 12: Class Diagram

4.2.9 E-R Diagram

taslak, çizim, diyagram, çizgi sanatı içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Figure 13: Entity-Relation Diagram

4.2.10 Logical Database

# metin, diyagram, ekran görüntüsü, çizgi içeren bir resim Açıklama otomatik olarak oluşturuldu

# Figure 14: Logical Database

4.2.11 Physical Database

metin, sayı, numara, yazı tipi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu Figure 15: Physical Database

# 5. IMPLEMENTATION

## 5.1 Tools, technologies and platforms used.

We used JavaScript as Programming language because of simplicity and performance. We used NodeJS as run environment and ExpressJS for backend side of project. This is because our project is not complex, we decided to keep code simple. We used MySQL as database for keep data tables organized and Sequelize for ORM. For frontend we used ReactJS to keep app dynamic and easy.

## 5.2 Algorithms

For Project our biggest challenge is organizing and formatting scrapped data. Each data scrapped has different format and type. For example, in different websites doctor specialities has same meaning but different format. To overcome this problem, we used Levenstein Distance Algorithm to manage similarity of specialties.

import { closest, distance } from"fastest-levenshtein";

constreduceSpecialities= (str) => {

  str=str

    .split(" ")

    .map((s) =>

      s!=="ve"

        ?s[0].toLocaleUpperCase("tr-TR") +

          s.substring(1).toLocaleLowerCase("tr-TR")

        :"ve"

    )

    .join(" ");

  str=str.replace(/Klinik|Medikal/g, "").trim();

  constdhRegex= /^İç Hastalık/g;

  consttrIndex=str.search(/-|Sağlığı Merkezi|Sorumlu Hekim|Doktoru/g);

  if (dhRegex.test(str)) return"Dahiliye";

  if (trIndex!==-1) returnstr.substring(0, trIndex).trim();

  returnstr.trim();

};

constlevensteinDistributor= (str) => {

  letspeciality=reduceSpecialities(str);

  constarr=hash[speciality[0]];

  returnclosest(speciality, arr);

};

# Figure 16: Levenstein Algorithm

This algorithm compares given specialty of doctor and stored specialties of previous doctors. Aim of this remove redundancy and keep data clean.

## 5.3 Standards

We decided to apply The Waterfall Methodology when developing our system. There are several reasons for that:

* Simple Project
* Limited Time and Budget
* Technology to be used stable and not likely to change.

We did apply OOP and SOLID principles while coding the project. Our code base is scalable, extendible, and simple. For frontend we did use HOC pattern to reduce redundant codes and increase performance.

## 5.4 Detailed description of the implementation (coding)

# metin, kadın, ekran görüntüsü, kişi, şahıs içeren bir resim Açıklama otomatik olarak oluşturuldu

# Figure 17: Doctors Screen

metin, ekran görüntüsü, web sitesi, çevrimiçi reklamcılık içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 18: Home Page Screen

# metin, harita, atlas, ekran görüntüsü içeren bir resim Açıklama otomatik olarak oluşturuldu

# Figure 19: Doctor Information Screen6. TESTING

We did use Jasmine Framework to test our app. We tested database initialization, route testing. For route testing our paramteres is just random ids.Here is sample code for this testing:

describe("Database test suite", () => {

  it("Database çalıştırır", async () => {

    awaitDatabase.initDatabase();

    expect(true).toBe(true);

    console.log("Databse Test Suite is Done and retruned OK");

  });

});

describe("Routes Test Suite", () => {

const randomNum = Math.floor(Math.random() \* 506))

  it("Return single Doctor", (done) => {

    supertest(Server.getApp())

      .get("/doctor/${randomNum}")

      .expect(201)

      .end((error) => (error?done.fail(error) :done()));

  });

  it("Gets Filters", (done) => {

    supertest(Server.getApp())

      .get("/filters/getfilters")

      .expect(201)

      .end((error) => (error?done.fail(error) :done()));

  });

});

# Figure 20: Testing

As test results we get all the functions are works just fine

# 7. USER GUIDE OF THE SYSTEM

# First, the user must have an internet connection. First, the user will log in to the website. The first thing he will see will be the introduction of the website. Here you can examine the services provided by the website. Then he/she should click on the Doctor link to see the doctors. On the new page that opens, he will be able to see the doctors that have already appeared. He can filter these doctors using the filters on the left side. To use the filtering, you must first click on specific filter items from the specified fields and click on the save button at the bottom. There is also a search box at the top right, if the user has a name in mind, they can search for it and select the doctors from the dropdown menu. If the user clicks on a doctor, they will be directed to their profile page. On this page, you can view detailed information about the doctor and get directions to the doctor's location with the map at the bottom. They can also find the doctor's phone number and email on the top right. 8. DISCUSSION

# Our project aims to make it easier for users to access some things in the field of health. This project will enable people to search and find health services globally as well as in Cyprus. Economically, since this project does not aim to make any profit, the user will be able to use this service without paying any money. In addition, it is normally more costly for the user to access the services promised by our project in other ways. Our project has basically found a solution to this problem.9. CONCLUSION

# In general, our project is a project that makes people's lives easier, saves effort and cost. With this project, we have learned to research and find solutions to people's needs in general.10. REFERENCES

# [www.draw.io.com](http://www.draw.io.com)

[www.react-dev.com](http://www.react-dev.com)

[www.nodeJS.com](http://www.nodeJS.com)

www.express.com

# APPENDICES

## A. Instructions for installing the system

1-Download NodeJS

2-Extraact zip to any folder

3- write npm run dev in both frontend and backend

4-write npm run dev in both frontend and backend

## B. Code for the system

## C. Other relevant material

Appendix

HTML - Hyper Text Markup Language

CSS - Cascading Style Sheet

JavaScript - High Level Programming

MySQL - Database

API - Application Programming Interface

NodeJS – Javascript Run Environment