

FACULTY OF ENGINEERING AND NATURAL SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING



MATCHMAKER
SOFTWARE DEVELOPEMNT REPORT

A GRADUATION PROJECT

SUBMITTED BY
GOKTURK RAMAZAN DOGAN

BACHELOR OF SCIENCE

JUNE 2022

PROGRAM: COMPUTER ENGINEERING

MATCHMAKER

A GRADUATION PROJECT

BY

GOKTURK RAMAZAN DOGAN

SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING OF

OKAN UNIVERSITY

BACHELOR OF SCIENCE

Approved by:

Member's
Name

Member's
Name

Member's
Name

Member's
Name

JUNE 2022

Program: COMPUTER ENGINEERING

ABSTRACT

MATCHMAKER

This project has two main objectives, the first of which is the problem of finding players in football match organizations, and the other is the lack of competition due to the fact that the established teams are not of equal strength. It provides the manager with the opportunity to create a player pool from the players whose information he/she will enter into the system, and gives the authority to give a score out of five to the players who take charge after the matches. In this way, both the player problem and the competition problem are solved.

Keywords:

- Football
- Team
- Player
- Facility
- Match
- Goal
- Point
- Vote
- Algorithm
- Goalkeeper
- Defence
- Midfielder
- Forward

CONTENTS

LIST OF TABLES	V
LIST OF FIGURES	VI
1. INTRODUCTION.....	1
1.1. TITLE.....	1
1.2. PROJECT DEFINITION	1
2. SOFTWARE REQUIREMENT SPECIFICATION	1
2.1. INTRODUCTION	1
2.2. GENERAL DESCRIPTION.....	1
2.3. PICTURE OF SYSTEM DATA FLOW	2
2.4. DATA TO BE STORED	3
2.5. FUNCTIONAL REQUIREMENTS	4
2.6. FUNCTIONAL BEHAVIORS	5
2.7. SYSTEM SPECIFICATION	9
2.8. NON-FUNCTIONAL ATTRIBUTES	10
2.9. VALIDATION CRETERIA	10
2.10. FUNCTION POINT ANALYS	10
3. SOFTWARE DESIGN DESCRIPTION.....	14
3.1. INTRODUCTION	14
3.2. DATA DESIGN.....	14
3.3. MODULE STRUCTURE	19
3.4. INTERFACE DESIGN	31
3.5. RECOMMENDATION	42
4. TEST PLAN AND TEST REPORT	42
4.1. ADDING PLAYER	42
4.2. ADDING FACILITY	43
4.3. ADDING MATCH.....	44
5. DISCUSSION AND CONCLUSION	46
5.1. SUCCESFULL ASPECTS	46
5.2. SEFICIENCIES	46
5.1. SUCCESTIONS	46
SOURCE	47
CV	48

LIST OF TABLES

Table 1	11
Table 2	12
Table 3	12
Table 4	13
Table 5	13
Table 6	15
Table 7	16
Table 8	16
Table 9	18
Table 10	18
Table 11	18
Table 12	19

LIST OF FIGURES

Figure 1	2
Figure 2	3
Figure 3	5
Figure 4	15
Figure 5	19
Figure 6	21
Figure 7	22
Figure 8	23
Figure 9	24
Figure 10	25
Figure 11	26
Figure 12	27
Figure 13	28
Figure 14	29
Figure 15	30
Figure 16-17	31
Figure 18-19	32
Figure 20-21	33
Figure 22-23	34
Figure 24-25	35
Figure 26	36
Figure 27-28	37
Figure 29-30-31-32	38
Figure 33-34-35	39
Figure 36-37-38	40
Figure 39-40-41	41
Figure 42	42
Figure 42-44	43
Figure 45-46-47	44
Figure 48	45

1. INTRODUCTION

1.1. TITLE

MatchMaker.

1.2. PROJECT DEFINITION

This project has two main objectives, the first of which is the problem of finding players in football match organizations, and the other is the lack of competition due to the fact that the established teams are not of equal strength. It provides the manager with the opportunity to create a player pool from the players whose information he/she will enter into the system, and gives the authority to give a score out of five to the players who take charge after the matches. In this way, both the player problem and the competition problem are solved.

2. SOFTWARE REQUIREMENTS SPECIFICATION

2.1 INTRODUCTION

2.1.1 Purpose of This Report

This report contains the software contents of the MatchMaker project.

2.1.2 Scope of This Report

This report describes the functional requirements and behaviour.

2.2 GENERAL DESCRIPTION

The MatchMaker project is a web-based project focused on the two most basic problems experienced in football competition organizations. The first of these is the problem of finding players. In order to solve this problem, the user who is given the administrator right will be able to create a pool of players by adding the information of the football players around him to the database of the project. The powers of the manager will also include functions such as adding matches within the project and adding potential facilities where the matches will be played. In this way, side problems such as finding a

facility and setting a match time will also be addressed. Another basic problem is that the teams formed are not equal to each other and therefore the competition and entertainment level of the competition is insufficient. To solve this problem, the project will include a voting system under the authority of the administrator. In this way, after the matches, the manager will be able to rate the players based on his observations in the match, and thus, while forming a team in the next match, he will be able to see their point match their played latly and create an organization with a higher level of competition by evenly distributing the players with high scores.

2.2.1. Critical Assumption

In this project, it is assumed that the people who are given the management will have an objective approach while voting. In case of abuse, no responsibility is accepted.

2.3. PICTURE OF SYSTEM, DATA FLOW

There are two types of users registered in our system. One of them is the administrative user, which has the authority to add, delete, vote and edit. Another is anonymous users who are not authorized to interact directly with the project's system and database. Anonymous users are not authorized to access the administration panel and the database, they can only see the interface designed for them and the reflection of the information in the database there. Their purpose is only to obtain information about the data in the system.. Context Diagram of Data Flow Diagram is shown in **Figure 1**.

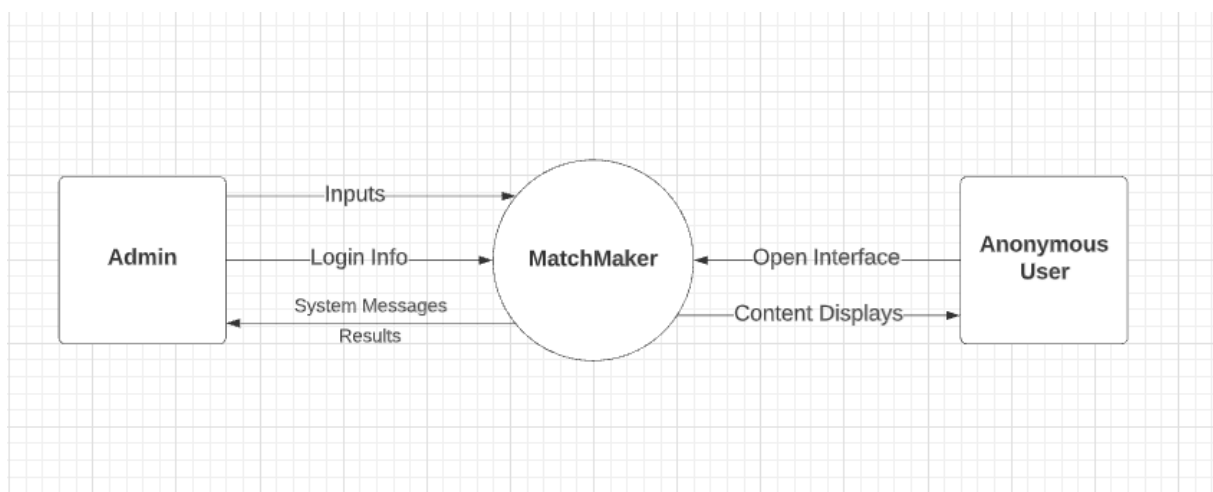


Figure 1.

When we look at the function diagram, the connection between the transactions and the database is shown in **Figure 2**. All inputs are taken from the administrator and forwarded to the database.

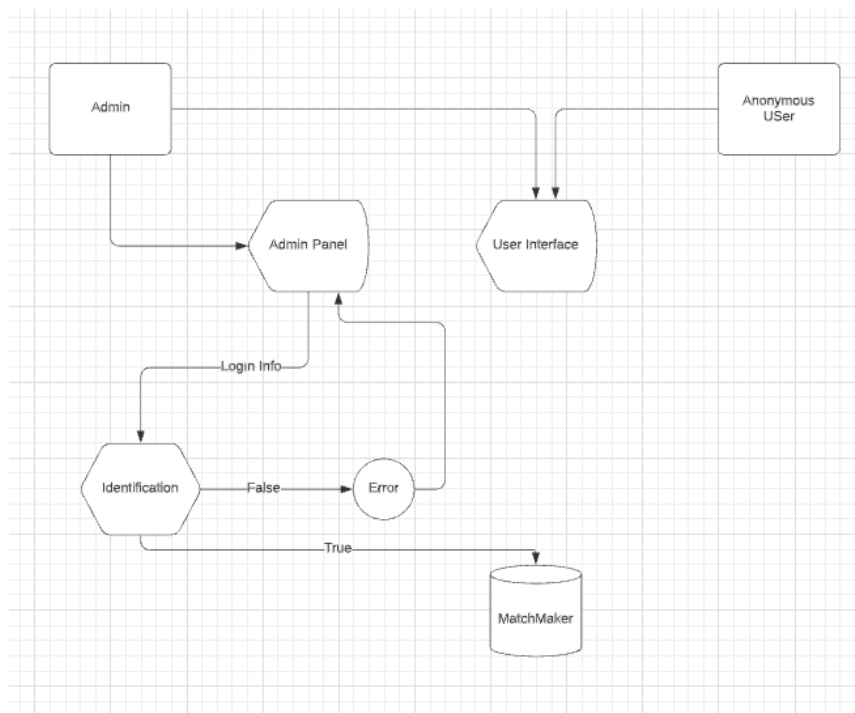


Figure 2.

2.4. DATA TO BE STORED

1. As stated in Chapter 2, this database will store data about players, matchers and facilities. These informations are football-related data such as the player's contact information, age, profile photo required for profile creation, the foot and position used by the player. For matches these are teams, date and time. For facilities these are location, phone number, facility name, field kind and field count.
2. In addition, in order for the system to fulfill its requirements, another type of information that the players must keep apart from their information is the information that will enable the user or users to perform the login function, such as the user name and password of the user or users who are given administrative authority.

2.5. FUNCTIONAL REQUIREMENTS

2.5.1 Functional Partitioning

The software of the project includes four different functions. All of these functions are available to the user with administrative authority. These are Add Player, Add Match, Add Facility, Edit Content, Delete Content and Vote.

2.5.2. Add Player

The user, who is given the administrator ID, can enter the information of the players he wants to add to the system through the inputs on the relevant page. The system will create a new row in the players table and save the data entered in the relevant columns.

2.5.3. Add Match

This function allows the user to create a match by selecting its date, time and facility and then add teams to that match via the required pages.

2.5.4. Add Facility

Through this function, the administrator user can enter the facilities where future matches will be organized into the system and ensure that they are recorded in the relevant table.

2.5.5. Edit Content

This function allows the admin user to update or edit the contents of data such as previously logged facilities and players. If it is used, it completes the editing process through the relevant inputs by calling the exact same insertion pages.

2.5.6. Delete Content

If the administrator decides to delete the previously entered content, he or she can go to the list page of the relevant content and click the button that enables the function to be deleted, thanks to this function.

2.5.7. Add Team

The administrator can add players in two teams to the matches via this function.

2.5.8. Vote

This function, which enables the manager to vote for the players who took part in the organized competition after the organized competitions, will only be accessible to the manager users.

2.6. FUNCTIONAL BEHAVIORS

2.6.1. Events, Actions and User Interface

In this section, the response of the program to all possible inputs and actions of the users as well as user interfaces will be described. The overall process flowchart is shown in **Figure 3**.

The screens that the user can perform actions are listed below

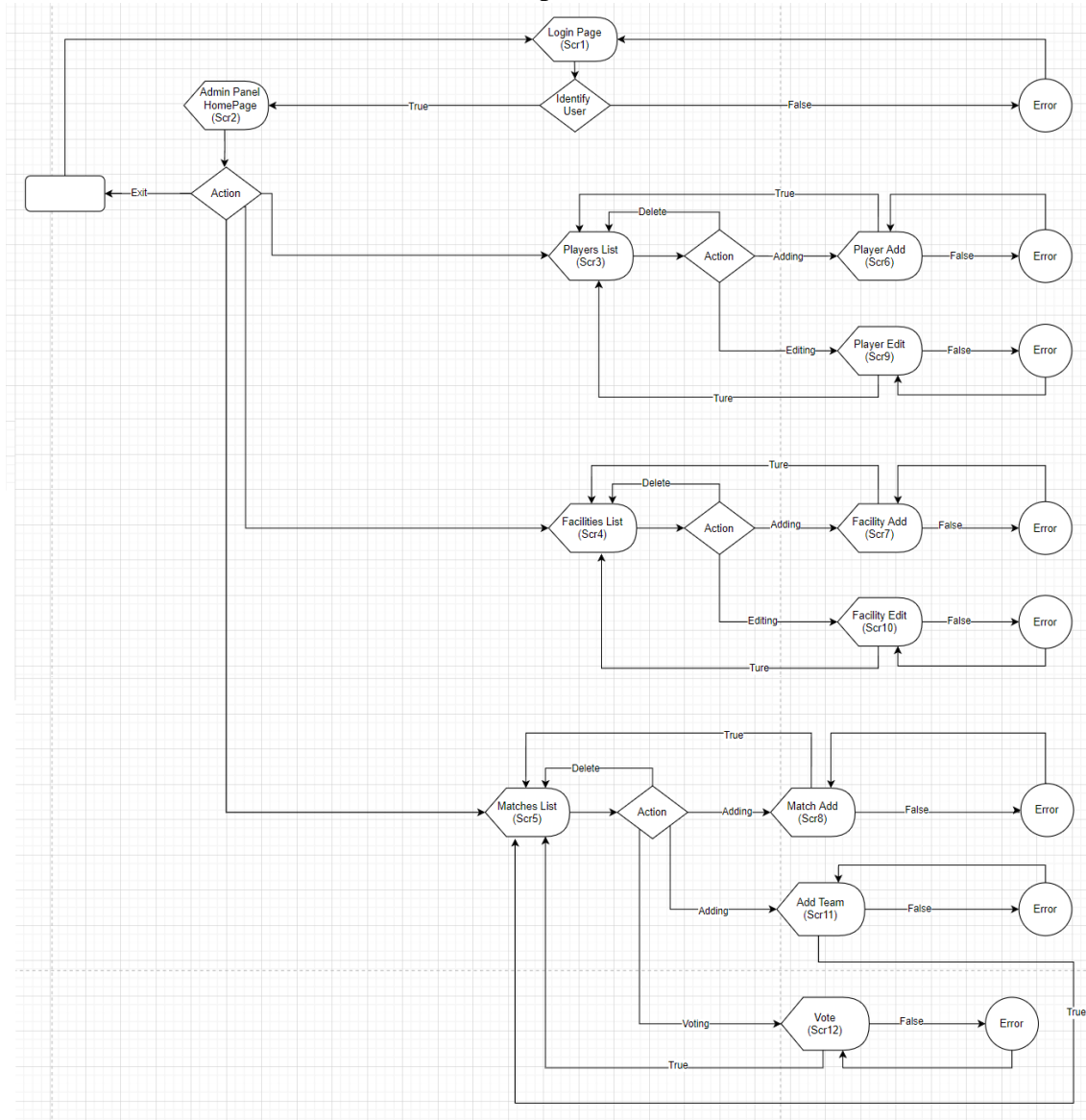


Figure 3.

2.6.1.1 Login Screen (Scr 1)

If the administration panel is opened, the user will first be greeted by a login form with two inputs, where user name and password entry will be provided.

1. If the login information entered by the user matches the data previously stored in the relevant table in the database, it will direct the user to the homepage of the Administration Panel (Scr 2).
2. If the login information entered by the user does not match the data in the table, a warning will appear that the information entered is incorrect and the user will be redirected to the login page (Src 1).

2.6.1.2 Admin Panel Home Page(Scr 2)

On this screen of the system, the administrative user is directed to the module adding page that creates a new category, to the list page that displays the contents of the previously created categories, and to the add, delete and edit pages via the list page.

1. By clicking on the name of the category for which the admin user wants to add, edit or delete content, the user can be directed to the page where the contents of the relevant category are listed.
2. The administrator can log out from the administration panel by clicking the 'Çıkış Yap' button at the bottom of the menu bar. As a result of this action, it is directed to the login page (Src 1) again.

2.6.1.3 Admin Panel 'Oyuncular' List Screen(Scr 3)

On this page of the administration panel, the administrator can view all the players in the category to be listed, together with the preview data.

1. Admin by clicking the 'Yeni ekle' button. can go to the relevant page (Scr6) to add a new player.
2. Admin can access the relevant edit page (Scr9) to edit the data of the relevant player by clicking the 'Düzenle' button.

3. To delete the player, the administrator user can delete the relevant player from the table in the database by clicking the 'Kaldır' button opposite the player. As a result, the administrator will be redirected to the relevant players list page (Scr 3) again.

2.6.1.4 Admin Panel 'Tesisler' List Screen(Scr 4)

On this page of the administration panel, the administrator can view all the facilities in the category to be listed, together with the preview data.

1. Admin by clicking the 'Yeni ekle' button. can go to the relevant page (Scr7) to add a new facility.
2. Admin can access the relevant edit page (Scr10) to edit the data of the relevant facility by clicking the 'Düzenle' button.
3. To delete the facility, the administrator user can delete the relevant facility from the table in the database by clicking the 'Kaldır' button opposite the facility. As a result, the administrator will be redirected to the relevant facility list page (Scr 4) again.

2.6.1.5 Admin Panel 'Maçlar' List Screen(Scr 5)

On this page of the administration panel, the administrator can view all the matches in the category to be listed, together with the preview data.

1. Admin by clicking the 'Yeni ekle' button. can go to the relevant page (Scr8) to add a new match.
2. The administrator can switch to the add team page (Scr11) to the relevant match by clicking the 'Takım A / Takım B' buttons.
3. By clicking the 'Oyla' button, the administrator can switch to the voting page (Scr12) of the relevant match and give points to the players who took part in the match and can enter the scores produced by the teams.
4. To delete the content, the administrator user can delete the relevant content from the table in the database by clicking the 'Kaldır' button opposite the match. As a result, the administrator will be redirected to the relevant list page (Src 5) again.

2.6.1.6 Admin ‘Oyuncular’ Adding Screen(Scr 6)

On this page, the administrator can enter new players into the system. The administrator cannot leave the inputs in the page blank due to the algorithm in which the function is written. These inputs are name, surname, location, phone number, picture, year of birth and the foot he used, respectively. As a result of the process, the user will be redirected to the players list (Src 3) page.

2.6.1.7 Admin ‘Tesisler’ Adding Screen(Scr 7)

On this page, the administrator can enter new facilities into the system. The administrator cannot leave the inputs in the page blank due to the algorithm in which the function is written. These inputs are facility name, address, site type, phone number, picture, and location, respectively. As a result of the process, the user will be redirected to the facility list (Src4) page.

2.6.1.8 Admin ‘Maçlar’ Adding Screen(Scr 8)

On this page, the administrator can enter new matches into the system. The administrator cannot leave the inputs in the page blank due to the algorithm in which the function is written. These inputs are date, address, time and facility, respectively. As a result of the process, the user will be redirected to the matches list (Src5) page.

2.6.1.9 Admin Panel Edit ‘Oyuncular’ Screen(Scr 9)

On this page of the user administration panel, the administrator can edit the data of the relevant player and save it back to the database in an updated way. As a result of this process, it will be redirected to the player list page (Src 3).

2.6.1.10 Admin Panel Edit ‘Tesisler’ Screen(Scr 10)

On this page of the user administration panel, the administrator can edit the data of the relevant facility and save it back to the database in an updated way. As a result of this process, it will be redirected to the facility list (Src 4) again.

2.6.1.11 Admin Panel ‘Maçlar’ Add Team Screen(Scr 11)

On this page, the administrator can add players in two teams to the matches and ensure that the data is transmitted to the table where the records of the matches are kept. As a result of this process, it will be redirected to the match list page (Src 5) again.

2.6.1.12 Admin Panel ‘Maçlar’ Vote Screen(Scr 12)

On this page, the administrator can give points out of 5 to the players who took part in a match that has been added and played before, and can save the scores produced by the teams to the system via inputs. As a result of this process, it will be redirected to the match list page (Src 5) again.

2.7. SYSTEM SPECIFICATION

2.7.1. Client

Although the MatchMaker project is a web-based project. It is recommended to run it in an HTML 5.0 and CSS 3.0 compatible browser. The project does not have any system requirements. For this reason, any device that has an internet connection and supports browser services can access this project without hardware barriers.

2.7.2. Server

The server to which the project is linked will be kept on the internet, so it is recommended that the users who will access it have a stable internet connection.

2.7.2.1. Required Software

The server is built on apache server with xampp control panel 3.4 version support. It works with phpMyAdmin version 4.2.

2.7.2.2. Required Hardware

There is no hardware requirement as the database information will be stored on the internet. 100mb of space on the relevant host will be sufficient.

2.8. NON-FUNCTIONAL ATTRIBUTES

2.8.1 Security

2.8.1.1 System Security

Database access will be provided only with the username and password to be created by the relevant host, and this information will only be given to administrative users. The default security practices of the phpMyAdmin system will remain in effect.

2.8.1.2 Network Security

This system, which will work in a local or global network environment, will host a .htaccess application with linking, so it will not receive any attack from the link input. The SSL certificate can then be applied to the site over the host.

2.8.1.3 User Security

Panel logins are made with user names and passwords. Each administrator ID has its own unique user name and password. It is the responsibility of the administrators to keep them confidential.

2.9. VALIDATION CRITERIA

Experts Database Software Development Project can be considered to be complete when the following conditions hold:

1. All functional requirements specified in Section 4 are met.
2. All user interfaces that are specified in Section 5 are complete.
3. The program responses to inputs described in Section 5 are as specified.

2.10. FUNCTION POINT ANALYSIS

2.10.1. Total Unadjusted Function Point Calculation

External Inputs

1. Login Screen **Simple**
2. HomePage **Simple**
3. List Screens **Average**

4. Edit Screens **Simple**
5. Adding Screens **Simple**
6. Vote Screen **Average**

External Outputs

1. HomePage **Simple**
2. List Screens **Average**

File Storage

1. Information Database **Complex**

External SW Interfaces

1. Data transfer interface between web browser and database **Complex**

External Query Types

1. Vote **Complex**

	Weighting Factor			
Measurement Parameter	Simple	Average	Complex	Total
External Inputs	4*4=	2*4=	0*6=	24
External Outputs	1*4=	1*5=	0*7=	9
File Storage	0*7=	0*10=	1*15=	15
External SW Interfaces	0*5=	0*7=	1*10=	10
Count Total				68

Table 1.

2.10.2. Adjusted Function Point Calculation

The function point count is adjusted for the complexity of the software by assessing each of the answers to the following questions on a scale of 0 to 5.

	F_i value
None	0
Slightly	1
Present	2
Medium level	3
Important	4
Vital	5

Table 2.

1. Data Communications	Vital	5
2. Distributed Processing	None	0
3. Performance Requirements	Important	4
4. Operational Configuration Load	None	0
5. Transaction Rate	Medium level	3
6. Online Data Input	Medium level	3
7. End User Quality	None	0
8. Online File Update	Important	4
9. Algorithmic Complexity	Slightly	1
10. Reusability	None	0
11. Ease of Installation	Slightly	1
12. Operational Ease	Slightly	1
13. Multi-site System	None	0
14. Maintainability	Medium level	3
	TOTAL	25

Table 3.

These 14 complexity adjustment values are summed to give the value of $\Sigma (F_i)$.

$$\Sigma (F_i) = 25$$

$$\begin{aligned} \text{Adjusted Function points} &= \text{Count Total} * (0.65 + 0.01 * \Sigma (F_i)) \\ &= 64 * 0.9 = \mathbf{58.5}. \end{aligned}$$

2.10.3. Lines of Code Estimation

The total adjusted function points came out to be 45.9 for the Experts' Database. For estimating lines of code we can use the **Table 1** below.

Language	(Albrecht 1983)	(Low & Jeffery 1990)
COBOL	110 LOC / FP	106 LOC / FP
PL / 1	65 LOC / FP	80 LOC / FP
4 GL	25 LOC / FP	40 LOC /FP

Table 4.

Estimated LOC for each FP.

Language	LOC(Albrecht 1983)	LOC(Low & Jeffery 1990)
COBOL	58.5*110=6435	58.5*106=6201
PL / 1	58.5*65=3803	58.5*80=4680
4 GL	58.5*25=1463	58.5*40=2340

Table 5.

LOC calculation.

For 4GL, the results are came up to be **1463 LOC** according to Albrecht 1983 and **2340 LOC** according to Low & Jeffery 1990.

3. SOFTWARE DESIGN DESCRIPTION

3.1. INTRODUCTION

3.1.1. Project Name:

MatchMaker.

3.1.2. Purpose of This Report:

This report contains information about the software design features and developability of the MatchMaker project.

3.1.3. View of The Report:

Data structure, module structure, administration panel admin interface and user interface designs are presented in this report. It must be used together with the requirements specification report.

3.1.4. Abbreviations and synonyms used:

Admin: Administrator.

Info: Information.

Int: User Interface.

UserID: User identification.

3.2. DATA DESIGN

3.2.1. Data Objects and Resultant Data Structures

The entity-relationship diagram of the MatchMaker is depicted in the below figure. The Matchmaker system has two data objects. one of them is the administrator data object and one is the login data object. The login data object contains the username and password information. If the entered data matches the data in the database, the login operation is successful. If it does not match, the login will fail.

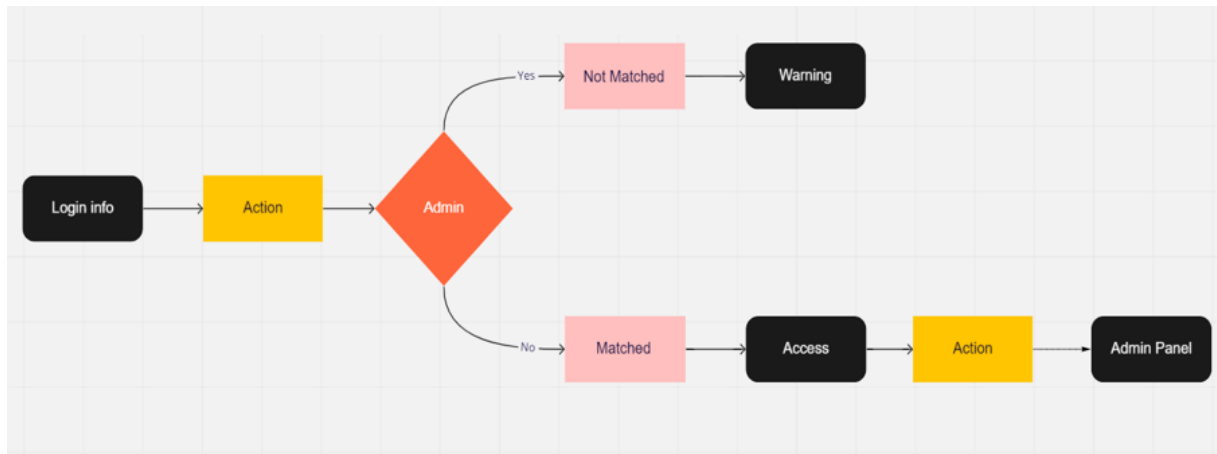


Figure 4.
Login Event MatchMaker

Based on the E-R model, following tables are created. The fields of tables, the type and primary key. All of them has created by admin manually from the database.

Fields	Type	Primary Key / Foreign Key	Required
ayarlar			
ID	Int(11)	Primary Key	
baslik	Varchar(160)		√
anahtar	Varchar(255)		√
aciklama	Varchar(255)		√
telefon	Varchar(50)		√
mail	Varchar(100)		√
adres	Varchar(255)		√
fax	Varchar(50)		√
url	Varchar(120)		√

Table 6.

Fields	Type	Primary Key / Foreign Key	Required
kullanici			
ID	Int(9)	Primary key	
adsoyad	Varchar(20)		√
kullanici	Varchar(20)		√
sifre	Varchar(20)		√
mail	Varchar(30)		√
tarih	Date		

Table 7.

Fields	Type	Primary Key / Foreign Key	Required
maclar			
ID	Int(11)	Primary key	
tarih	Varchar(55)		√
saat	Varchar(55)		√
tesis	Varchar(120)		√
takimakaleci	Varchar(500)		
takimadefans1	Varchar(500)		
takimadefans2	Varchar(500)		
takimadefans3	Varchar(500)		
takimaortasaha1	Varchar(500)		
takimaortasaha2	Varchar(500)		
takimaforvet	Varchar(500)		
takimbkaleci	Varchar(500)		
takimbdefans1	Varchar(500)		
takimbdefans2	Varchar(500)		
takimbdefans3	Varchar(500)		
takimbortasaha1	Varchar(500)		

takimbortasaha2	Varchar(500)		
takimbforvet	Varchar(500)		
takimakalecipuan	Int(11)		
takimadefans1pu an	Int(11)		
takimadefans2pu an	Int(11)		
takimadefans3pu an	Int(11)		
takimaortasaha1p uan	Int(11)		
takimaortasaha2p uan	Int(11)		
takimaforvetpuan	Int(11)		
takimbkalecipua n	Int(11)		
takimbdefans1pu an	Int(11)		
takimbdefans2pu an	Int(11)		
takimbdefans3pu an	Int(11)		
takimbortasaha1 puan	Int(11)		
takimbortasaha2 puan	Int(11)		
takimbforvetpua n	Int(11)		
takimaskor	Int(55)		
takimbskor	Int(55)		

Table 8.

Fields	Type	Primary Key / Foreign Key	Required
mevki			
ID	Int(11)	Primary key	
mevki	Varchar(120)		√
url	Varchar(500)		√

Table 9.

Fields	Type	Primary Key / Foreign Key	Required
mevki			
ID	Int(11)	Primary key	
isim	Varchar(255)		√
soyisim	Varchar(255)		√
mevki	Varchar(255)		√
telefonnumarasi	Varchar(255)		√
ayak	Varchar(255)		√
resim	Varchar(500)		√
dogumtarihi	Varchar(255)		√
mevkiid	Int(11)		√

Table 10.

Fields	Type	Primary Key / Foreign Key	Required
puan			
ID	Int(11)	Primary key	
mevki	Varchar(120)		√
url	Varchar(500)		√

Table 11.

Fields	Type	Primary Key / Foreign Key	Required
tesisler			
ID	Int(11)	Primary key	
tesisadi	Varchar(255)		√
sahasayisi	Varchar(255)		√
telefon	Varchar(30)		√
sahatipi	Varchar(20)		√
resim	Varchar(255)		√
adres	Varchar(255)		√
konum	Varchar(500)		√

Table 12.

3.3. MODULE STRUCTURE(FOR ADMIN PANEL)

3.3.1. Module Map:

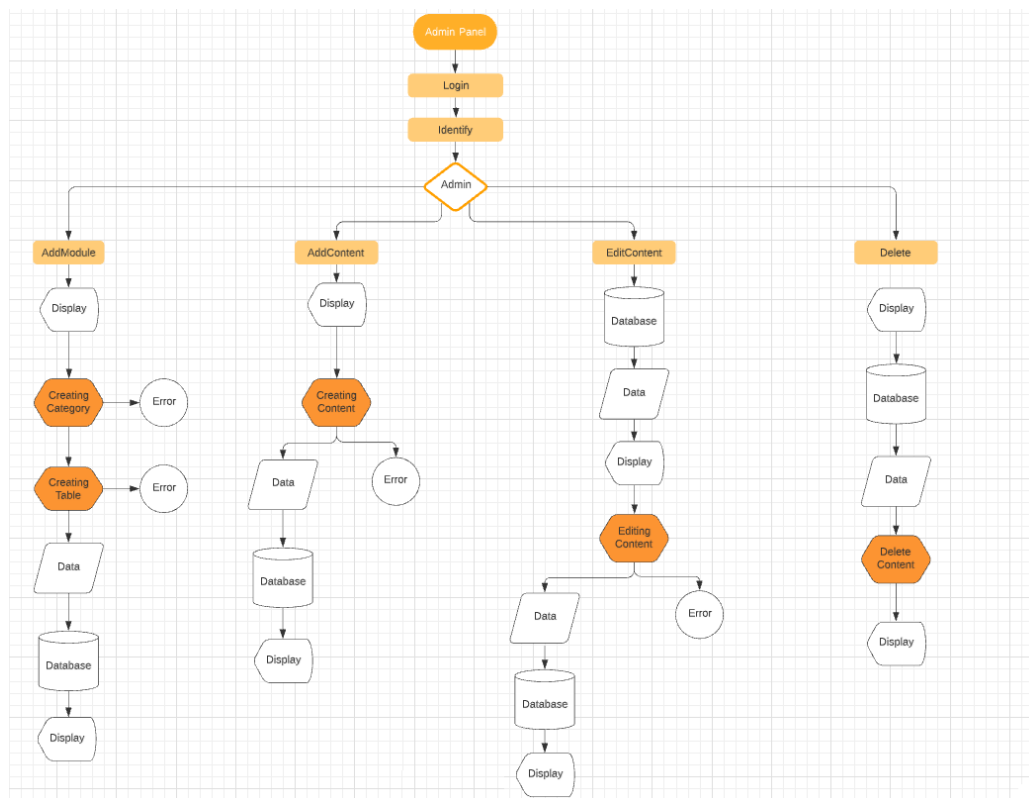


Figure 5.

3.3.2. Login:

Function : Program entry, gets name and password of the user and passes to Identify module .

Called by : Main program entry, called firstly after running the AdminPanel.

Calls : Identify module.

Inputs : Name and password information from user.

Outputs : Name and password information from user passes to Identify module.

Algorithm : Display Login Page (see Int.1.) shown in part 5-interface design.

Get name and password of user as strings from login form at the page of Login.

Call Identify module.

3.3.3. Identify:

Function : Gets name and password of the user (see Int.1.) shown in part 5-interface design. Checks their validity from Admin table from database.

Called by : Login Module.

Calls : Error Module, Admin Panel Homepage.

Inputs : Username and password information.

Outputs : If login is success Refreshes the page and goes to Main page of Admin Panel (see Int. 2.). If it is not, displays (see Int. 6-1-1. or Int. 6-1-2.) shown in part 5-interface design.

Algorithm :

```
if($_POST){
    if(!empty($_POST["username"]) && (!empty($_POST["password"]))) {
        $kullanici=$VT->filter($_POST["username"]);
        $sifre=md5($VT->filter($_POST["password"]));
        $kontrol=$VT->getdata("kullanici", "WHERE username=? AND password=?", array($kullanici,$sifre), "ORDER BY ID ASC", 1);
        if($kontrol!=false){
            $_SESSION["username"]=$kontrol[0]["username"];
            $_SESSION["namesurname"]=$kontrol[0]["namesurname"];
            $_SESSION["mail"]=$kontrol[0]["mail"];
            $_SESSION["ID"]=$kontrol[0]["ID"];
            ?>

            <meta http-equiv="refresh" content="0;url=?=SITE?>" />
            <?php
            exit();
        }
        else{
            echo '<div class="alert alert-danger">Yanlis Kullanici Adi veya Sifre.</div>';
        }
    }
    else{
        echo '<div class="alert alert-danger">Bos biraktiginiz Yerleri Doldurunuz.</div>';
    }
}
```

Figure 6.

3.3.4. Addplayers:

Function : This module saves the player information entered by the administrator on the relevant form page (see Int. 3-2.) in the 'oyuncular' table. shown in part 5-interface design.

Called by : Identify module.

Calls : Error Module.

Inputs : Information of the player to be created.

Outputs : If the add operation is successful displays (see Int. 6-2-1.) and switches to players-list (see Int. 3-1.). But if the add operation is not successful displays (see Int. 6-3.) shown in part 5-interface design.

Algorithm :

```
include "../DB.php";
$db=new DB();
if(isset($_POST["submit"])){
    $upl='../matchmaker/uploads/';
    $upload = $upl . basename($_FILES['resim']['name']);
    if (move_uploaded_file($_FILES['resim']['tmp_name'], $upload)) {
        $params=array($_POST ['isim'],$_POST ['soyisim'],$_POST ['mevki'],$_POST ['mevkiid'],$_POST ['telefonnumarasi'],$upload,$_POST ['dogumtarihi'],$_POST ['ayak']);
        if($db->insert('oyuncular', 'isim,soyisim,mevki,mevkiid,telefonnumarasi,resim,dogumtarihi,ayak', $params, 8))
        {
            ?><div class="alert alert-success">Isleminiz Basari Ile Kaydedildi.</div><?php ?>
            <meta http-equiv="refresh" content="2;url=oyuncular-liste.php">
        <?php
    }
    else{
        ?>
        <div class="alert alert-danger">Isleminiz Kaydedilirken Sorun Olustu.</div>
        <?php
    }
}
else{
    ?>
    <div class="alert alert-danger">Isleminiz Kaydedilirken Sorun Olustu.</div>
    <?php
}
}
```

Figure 7.

3.3.5. Addfacilities:

Function : This module by taking the information of the facility to be entered from the form on the page (see Int. 4-2.) by admin. records in the corresponding columns in the 'tesisler' table.

Called by : Identify module.

Calls : Editfacilities Module, Delete Module, Error Module.

Inputs : Information of the faicity to be created.

Outputs : If the save operation is successful display (see Int. 6-2-1.) and switches to facility-list page (see Int. 4-1.) in part-5. But if the save is not successful displays (see Int. 6-3.) in part-5.

Algorithm : Display Addfacility page (see Int. 4-2.) in part-5.

```
include "../../DB.php";
$db=new DB();
if(isset($_POST["submit"])){
    $upl="../../matchmaker/uploads/";
    $upload = $upl . basename($_FILES['resim']['name']);
    if (move_uploaded_file($_FILES['resim']['tmp_name'], $upload)) {
        $params=array($_POST ['tesisadi'],$_POST ['adres'],$_POST ['sahatipi'],$_POST ['telefon'],$upload,$_POST ['sahasayisi'],$_POST ['konum']);
        if($db->insert('tesisler', 'tesisadi,adres,sahatipi,telefon,resim,sahasayisi,konum', $params, 7))
        {
            ?><div class="alert alert-success">İşleminiz Başarılı Olarak Kaydedildi.</div>
            <?php ?>
            <meta http-equiv="refresh" content="2;url=tesisler-liste.php">
        <?php
        }
        else{
            ?>
            <div class="alert alert-danger">İşleminiz Kaydedilirken Sorun Oluşturdu.</div>
            <?php
        }
    }
}
```

Figure 8.

3.3.6. Addmatches:

Function	: This module saves the match data received by the admin from the form inputs on the match add page to the 'maclar' table.
Called by	: Identify module.
Calls	: Editmatch Module, Delete Match Module, Error Module, Addteam Module, Vote Module.
Inputs	: Information of the match to be created.
Outputs	: If the save operation is successful display (see Int. 6-2-1.) and switches to match-list page (see Int. 5-1.) in part-5. But if the save is not successful displays (see Int. 6-3.) in part-5.
Algorithm	: Display Addmatches page (see Int. 5-2.) in part-5.

```
include("../getdata.php");
include "../DB.php";
$db=new DB();
if(isset($_POST["submit"])){

$params=array($_POST ['tarih'],$_POST ['saat'],$_POST ['tesis']);
    if($db->insert('maclar', 'tarih,saat,tesis', $params, 3))
    {
        ?><div class="alert alert-success">İşleminiz Başarılı İle Kaydedildi.</div>
        <?php ?>
        <meta http-equiv="refresh" content="2;url=maclar-liste.php">
    <?php
    }

}
```

Figure 9.

3.3.7. Editplayers:

- Function : It allows editing the data of the players added to the related table with the Addplayers algorithm.
- Called by : Identify Module, Addplayers Module.
- Calls : Error Module, Delete Module.
- Inputs : Replacing the data of the relevant player with the data to be updated.
- Outputs : It displays (see Int. 6-2-1.) or (see Int. 6-3.) in part-5.
- Algorithm : After the editing process is finished displays (see Int. 6-2-1.) if the proccess is success. But if it is not displays (see Int. 6-3.) in part-5.

```
include "../../getdata.php";
include "../../DB.php";
include "../../class/vt.php";

if(isset($_POST["guncelle"])) {
    $id= $_POST["id"];
    $update=$vt-> prepare("UPDATE oyuncular set isim = :isim, soyisim = :soyisim,
    mevki = :mevki, telefonnumarasi = :telefonnumarasi, dogumtarihi = :dogumtarihi, ayak = :ayak where id = :id");
    $oyuncular = $update -> execute(array(
    'isim' => $_POST["isim"],
    'soyisim' => $_POST["soyisim"],
    'mevki' => $_POST["mevki"],
    'telefonnumarasi' => $_POST["telefonnumarasi"],
    'dogumtarihi' => $_POST["dogumtarihi"],
    'ayak' => $_POST["ayak"],
    'id' =>$_GET["id"]

    ));
    ?><div class="alert alert-success">İşleminiz Başarı ile Kaydedildi.</div><?php
    ?>
    <meta http-equiv="refresh" content="2;url=oyuncular-liste.php">
<?php
}
```

Figure 10.

3.3.8. Editfacilities:

Function	: It allows editing the data of the facility added to the related table with the Addfacilities algorithm.
Called by	: Identify Module, Addfacilities Module.
Calls	: Error Module, Delete Module.
Inputs	: Replacing the data of the relevant facility with the data to be updated.
Outputs	: It displays (see Int. 6-2-1.) or (see Int. 6-3.) in part-5.
Algorithm	: After the editing process is finished displays (see Int. 6-2-1.) if the proccess is success. But if it is not displays (see Int. 6-3.) in part-5.

```
if(isset($_POST["guncelle"])) {  
    $id= $_POST["id"];  
    $update=$vt-> prepare("UPDATE tesisler set tesisadi = :tesisadi, sahasayisi = :sahasayisi,  
    telefon = :telefon, sahatipi = :sahatipi, adres = :adres, konum = :konum where id = :id");  
    $tesisler = $update -> execute(array(  
'tesisadi' => $_POST["tesisadi"],  
'sahasayisi' => $_POST["sahasayisi"],  
'telefon' => $_POST["telefon"],  
'sahatipi' => $_POST["sahatipi"],  
'adres' => $_POST["adres"],  
'konum' => $_POST["konum"],  
'id' =>$_GET["id"]  
  
    ));  
    ?><div class="alert alert-success">İşleminiz Başarılı file Kaydedildi.</div><?php  
    ?>  
    <meta http-equiv="refresh" content="2;url=tesisler-liste.php">  
<?php  
}
```

Figure 11.

3.3.9. Addteam:

Function	: This module gives the authority to add the players as teams to the positions where they will be assigned by the manager and adds this data to the columns of the relevant match in the 'maclar' table.
Called by	: Addmatches Module.
Calls	: Error Module, Delete Module, Vote Module.
Inputs	: Selection of players from the relevant select inputs by the administrator
Outputs	: It displays (see Int. 6-2-1.) or (see Int. 6-3.) in part-5.
Algorithm	: After the adding process is finished displays (see Int. 6-2-1.) if the process is success. But if it is not displays (see Int. 6-3.) in part-5.

```
if(isset($_POST["guncelle"])) {  
    $id= $_POST["id"];  
    $update=$vt-> prepare("UPDATE maclar set takimakaleci = :takimakaleci, takimadefans1 = :takimadefans1,  
    takimadefans2 = :takimadefans2, takimadefans3 = :takimadefans3, takimaortasaha1 = :takimaortasaha1,  
    takimaortasaha2 = :takimaortasaha2, takimaforvet = :takimaforvet where id = :id");  
    $maclar = $update -> execute(array(  
        'takimakaleci' => $_POST["takimakaleci"],  
        'takimadefans1' => $_POST["takimadefans1"],  
        'takimadefans2' => $_POST["takimadefans2"],  
        'takimadefans3' => $_POST["takimadefans3"],  
        'takimaortasaha1' => $_POST["takimaortasaha1"],  
        'takimaortasaha2' => $_POST["takimaortasaha2"],  
        'takimaforvet' => $_POST["takimaforvet"],  
        'id' =>$_GET["id"]  
    ));  
    ?><div class="alert alert-success">İşleminiz Başarılı file Kaydedildi.</div><?php  
    ?>  
    <meta http-equiv="refresh" content="2;url=maclar-liste.php">  
<?php  
}
```

Figure 12.

3.3.10. Vote:

Function	: This module records the score entered by the administrator and the performance information of the players out of 5 in the relevant columns in the 'maclar' table.
Called by	: Addmatches Module, Addteam Module.
Calls	: Error Module, Delete Module.
Inputs	: The score produced by the teams entered by the manager and the performance score shown by the players.
Outputs	: It displays (see Int. 6-2-1.) or (see Int. 6-3.) in part-5.
Algorithm	: After the voting process is finished displays (see Int. 6-2-1.) if the process is success. But if it is not displays (see Int. 6-3.) in part-5.

```
if(isset($_POST["guncelle"])) {
    $id= $_POST["id"];
    $update=$vt-> prepare("UPDATE maclar set takimakalecipuan = :takimakalecipuan, takimadefans1puan = :takimadefans1puan,
    takimadefans2puan = :takimadefans2puan, takimadefans3puan = :takimadefans3puan, takimaortasaha1puan = :takimaortasaha1puan,
    takimaortasaha2puan = :takimaortasaha2puan, takimaforvetpuan = :takimaforvetpuan, takimbkalecipuan = :takimbkalecipuan,
    takimbdefans1puan = :takimbdefans1puan, takimbdefans2puan = :takimbdefans2puan, takimbdefans3puan = :takimbdefans3puan,
    takimbortasaha1puan = :takimbortasaha1puan, takimbortasaha2puan = :takimbortasaha2puan, takimbforvetpuan = :takimbforvetpuan,
    takimaskor = :takimaskor, takimbskor = :takimbskor
    where id = :id");
    $maclar = $update -> execute(array(
    'takimakalecipuan' => $_POST["takimakalecipuan"],
    'takimadefans1puan' => $_POST["takimadefans1puan"],
    'takimadefans2puan' => $_POST["takimadefans2puan"],
    'takimadefans3puan' => $_POST["takimadefans3puan"],
    'takimaortasaha1puan' => $_POST["takimaortasaha1puan"],
    'takimaortasaha2puan' => $_POST["takimaortasaha2puan"],
    'takimaforvetpuan' => $_POST["takimaforvetpuan"],
    'takimbkalecipuan' => $_POST["takimbkalecipuan"],
    'takimbdefans1puan' => $_POST["takimbdefans1puan"],
    'takimbdefans2puan' => $_POST["takimbdefans2puan"],
    'takimbdefans3puan' => $_POST["takimbdefans3puan"],
    'takimbortasaha1puan' => $_POST["takimbortasaha1puan"],
    'takimbortasaha2puan' => $_POST["takimbortasaha2puan"],
    'takimbforvetpuan' => $_POST["takimbforvetpuan"],
    'takimaskor' => $_POST["takimaskor"],
    'takimbskor' => $_POST["takimbskor"],
    'id' =>$_GET["id"],
    ));
    ?><div class="alert alert-success">İşleminiz Başarılı İle Kaydedildi.</div><?php
    ?>
    <meta http-equiv="refresh" content="2;url=maclar-liste.php">
<?php
}
```

Figure 13.

3.3.11. Delete:

Function : Deletes all data of the related content and itself from the related table.

Called by : Addplayers Module, Addfacilities Module, Addmatches Module.

Calls : None

Input : Admin click.

Outputs : None

Algorithm :

```
if(!empty($_GET["tablo"]) && !empty($_GET["ID"])){

    $tablo=$VT->filter($_GET["tablo"]);
    $ID=$VT->filter($_GET["ID"]);
    $kontrol=$VT->getdata("moduller","WHERE tablo=? AND durum=?",array($tablo,1),"ORDER BY ID ASC",1);
    if($kontrol!=false){
        $veri=$VT->getdata($kontrol[0]["tablo"],"WHERE ID=?",array($ID),"ORDER BY ID ASC",1);
        if($veri!=false){
            $sil=$VT->SorguCalistir("DELETE FROM ".$tablo, "WHERE ID=?",array($ID),1);
            ?>
            <meta http-equiv="refresh" content="0;url=<?=SITE?>index.php?page=liste&tablo=<?=$kontrol[0]["tablo"]?>">
            <?php
            }
            else{
            ?>
            <meta http-equiv="refresh" content="0;url=<?=SITE?>index.php?page=liste&tablo=<?=$kontrol[0]["tablo"]?>">
            <?php
            }
        }
    }
    else{
        ?>
        <meta http-equiv="refresh" content="0;url=<?=SITE?>">
        <?php
    }
}
else{
    ?>
    <meta http-equiv="refresh" content="0;url=<?=SITE?>">
    <?php
}
?>
```

Figure 14.

3.3.12. Error:

Function	: Displays a warning when algorithm return false value.
Called by	: Identify module, Addplayers Module, Addfacilities Module, Addmatches Module, Editplayers Module, Editfacilities Module, Addteam Module, Vote Module.
Calls	: none
Input	: Admin click.
Outputs	: Error Messages which created by admin.
Algorithm	: (see Int.6-1-1.),(see Int.6-1-2.),(see Int.6-1-3.).

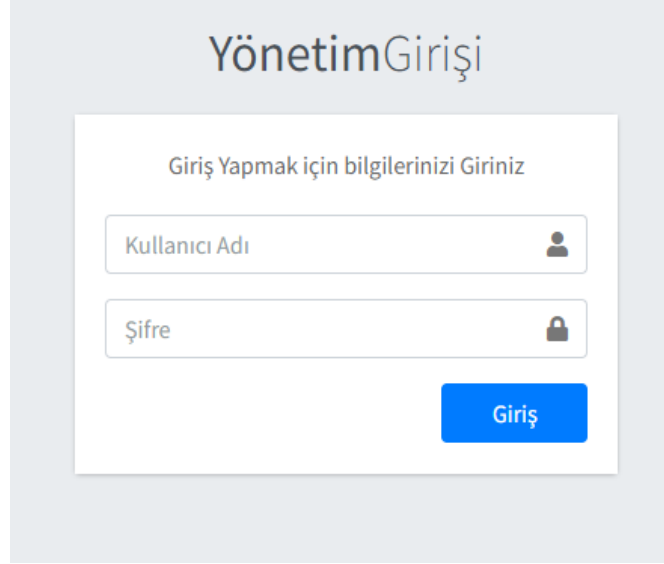
```
if($_SAMPLEMETHOD){  
    $sampleparameter=$DataBaseClass->SampleFunction();  
    if($sampleparameter!=false){  
        echo '<div class="alert alert-success"> Success Message .</div>';  
        ?>  
        <meta http-equiv="refresh" content="2;url=<?=SITE?>">  
        <?php  
        }  
    }  
    else{  
        echo '<div class="alert alert-success">Error Message</div>';  
    }  
}
```

Figure 15.

3.4. INTERFACE DESIGN

3.4.1. ADMIN PANEL

Int. 1. Login Page.



The login page features a light gray background. At the top center, the title "Yönetim Giriş" is displayed in a large, dark gray font. Below the title, a white rectangular box contains the login form. Inside this box, the text "Giriş yapmak için bilgilerinizi giriniz" is centered. There are two input fields: the first is labeled "Kullanıcı Adı" (Username) and the second is labeled "Şifre" (Password). Each field has a small icon on the right (a person icon for the username and a lock icon for the password). A blue button labeled "Giriş" (Login) is positioned at the bottom right of the input fields.

Figure 16.

Int. 2. Default HomePage.

YÖNETİM PANELİ



Figure 17.

Int. 3-1. PlayersList Page.

Oyuncular Listesi								
Show 10 entries		Search: <input type="text"/>						
#	Adı	Soyadı	Mevkisi	Kullandığı Ayak	Yaşı	İşlem		
1	Göktürk	DOĞAN	Forvet	Sağ	24	Düzenle	Kaldır	
2	Oğuzhan	ŞENOKUR	Kaleci	Sol	24	Düzenle	Kaldır	
3	Burak	ÖZER	Defans	Sağ	24	Düzenle	Kaldır	
4	Mustafa	ÖZKAN	Defans	Sol	25	Düzenle	Kaldır	
5	Melih	ERTÜRK	Ortasaha	Sağ	22	Düzenle	Kaldır	
6	Hakan	KESKİN	Defans	Sağ	23	Düzenle	Kaldır	
7	Uğur	ONAR	Forvet	Sağ	23	Düzenle	Kaldır	
8	Halil	REİSOĞLU	Ortasaha	Sağ	23	Düzenle	Kaldır	
9	Doğancan	MAVİDENİZ	Ortasaha	Sağ	22	Düzenle	Kaldır	

Figure 18.

Int. 3-2. Addplayers Page.

Oyuncu Ekle

İsim

İsim...

Soyisim

Soyisim...

Mevki

Mevki ID

(Kaleci=1, Defans=2, Ortasaha=3, Forvet=4)

Açıklama

Telefon Numarası...

Resim

Dosya Seç

Dosya seçilmedi

Doğum Yılı

Kullandığı Ayak

Gönder

Figure 19.

Int. 3-3. Player-Edit Page.

Oyuncu Düzenle

İsim

Soyisim

Mevki

Telefon Numarası

Doğum Yılı

Kullandığı Ayak

Güncelle

Figure 20.

Int. 4-1. Facilities-List Page.

Tesisler Listesi						
Show 10 entries		Search: <input type="text"/>				
#	Tesis Adı	Saha Sayısı	Telefon	Saha Tipi	Adres	İşlem
1	AydınOral	3	05333012072	Kapalı	Çamlık, Cahit Zarifoğlu Cd. No:7, 34912 Pendik/İstanbul	<button>Düzenle</button> <button>Kaldır</button>
2	Çamlık	3	(0216)5865775	Kapalı	Çamlık, Erguvan Sokağı No:2, 34912 Pendik/İstanbul	<button>Düzenle</button> <button>Kaldır</button>
3	Arena	2	05300611717	Kapalı	Harmandere, Ankara Cd., 34912 Kurna/Pendik/İstanbul	<button>Düzenle</button> <button>Kaldır</button>
4	İrem	1	(0216)4826061	Açık	Yenişehir, Mustafa Akyol Sokağı, 34912 Pendik/İstanbul	<button>Düzenle</button> <button>Kaldır</button>
#	Adı	Soyadı	Mevkisi	Kullandığı Ayak	Doğum Tarihi	İşlem

Figure 21.

Int. 4-2. Facilityadd Page.

Tesis Ekle

Tesis Adı

Adres

Saha Tipi

Telefon Numarası

Resim

Saha Sayısı

Konum

Gönder

Figure 22.

Int. 4-3. Facility-edit Page.

Tesis Düzenle

Tesis Adı

Adres

Saha Tipi

Telefon Numarası

Saha Sayısı

Konum

Gönder

Figure 23.

Int. 5-1. Matches-list Page.


Maç Listesi							
Show 10 entries				Search: <input type="text"/>			
#	Tarih	Saat	Tesis	Takım A	Takım B	İşlem	
1	2022-05-02	22:30	Arena	İbrahim 4 Ali 3 Mustafa 1 Alican 1 Ömer 4 Efe 5 Göktürk 3	Hakan 4 Berke 4 Emre 2 Ozan 4 Serhat 2 Feyzullah 4 Ali 2	<button>Takım A</button> <button>Takım B</button> <button>Oyla</button> <button>Kaldır</button>	
2	2022-05-07	23:00	İrem	Oğulcan 1 Selçuk 3 Ertuğrul 5 Cemal 2 Kadir 3 Tayyip 4 Göktürk 5	Hakan 4 Ozan 5 Emre 2 Alican 2 Feyzullah 5 Serhat 1 Ali 2	<button>Takım A</button> <button>Takım B</button> <button>Oyla</button> <button>Kaldır</button>	
#	Tarih	Saat	Tesis	Takım A	Takım B	İşlem	

Figure 24.


Int. 5-2. Addmatch Page.

Maç Ekle

Tarih

gg.aa.yyyy 

Saat

--:-- 

Tesis

Gönder

Figure 25.



Figure 26.

Int. 5-4. Vote Page.

TAKIM A Skoru Giriniz

Ibrahim

Ali Mustafa Alican

Ömer Efe

Gökürk

Ali

Serhat Feyzullah

Berke Emre Ozan

Hakan

TAKIM B Skoru Giriniz

Gönder

Figure 27.

Int. 6-1-1. Error.

Boş Bıraktığınız Yerleri Doldurunuz.

Figure 28.

Int. 6-1-2. Error.

Yanlış Kullanıcı Adı veya Şifre.

Figure 29.

Int. 6-1-3. Error.

İşleminiz Kaydedilirken Sorun Oluştı.

Figure 30.

Int. 6-2-1. Success.

İşleminiz Başarı İle Kaydedilmiştir.

Figure 31.

Int. 7. Menu.

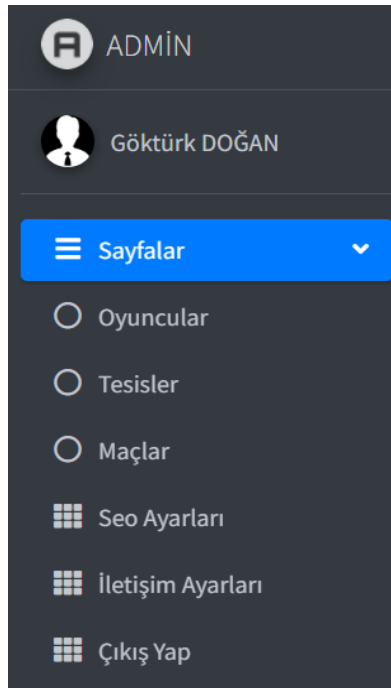


Figure 32.

3.4.2 USER INTERFACE

Int. 8. HomePage.

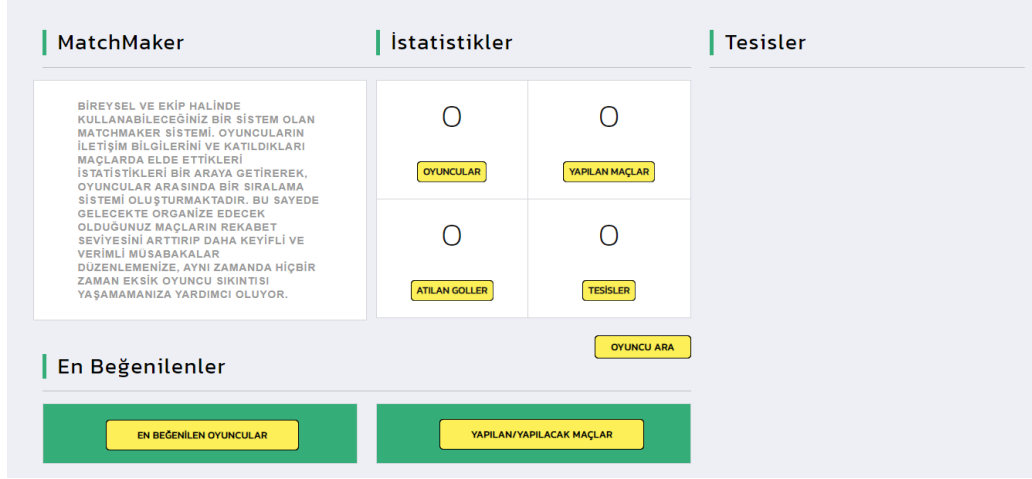


Figure 33.

Int. 9. Players Page.


Oyuncu Sıralaması						
OYUNCU ARAMA						
#	AYAK	OYUNCU	MEVKİ	YAŞ	PUAN	
1	Sol	 Örnek Oyuncu	Kaleci	27	43	

Figure 34.

Int. 9-1. Player Detail Page.



Örnek		Oyuncu Pozisyonu	
	ADI: Örnek SOYADI: Oyuncu MEVKİSİ: Kaleci TELEFON NUMARASI: 05555555555 KULLANDIĞI AYAK: Sol YAŞI: 27 PUANI: 43		

Figure 35.

Int. 10-1 Facilities Page.





Tesisler					
TESİS ARAMA					
#	TESİS ADI		SAHA TİPİ	SAHA SAYISI	DETAY
1	AydınOral		Kapalı	3	Görüntüle
2	Çamlık		Kapalı	3	Görüntüle
3	Arena		Kapalı	2	Görüntüle
4	İrem		Açık	1	Görüntüle

Figure 36.

Int. 10-2 Facility-Detail Page.



Çamlık		Konum Bilgisi	
	<p>TESİS ADI: Çamlık</p> <p>SAHA SAYISI: 3</p> <p>TELEFON: (0216)5865775</p> <p>SAHA TİPİ: Kapalı</p> <p>ADRES: Çamlık, Erguvan Sokakı No:2, 34912 Pendik/İstanbul</p>		

Figure 37.

Int. 11-1 Matches Page.

Maçlar				
MAÇ ARAMA				
#	TARİH	SAAT	TESİS	DETAY
1	2022-05-02	22:30	Arena	Görüntüle
2	2022-05-07	23:00	İrem	Görüntüle

Figure 38.

Int. 11-2 Match-Detail Page.

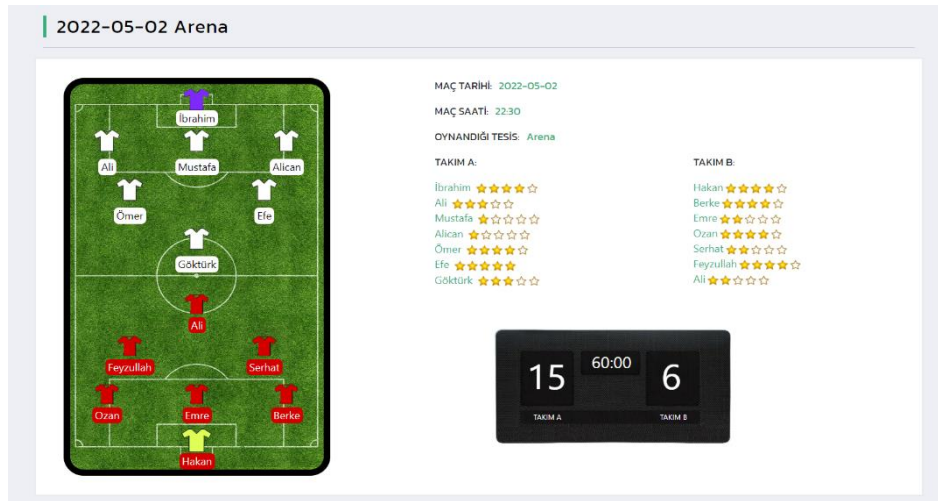


Figure 39.

Int. 12 Menu.

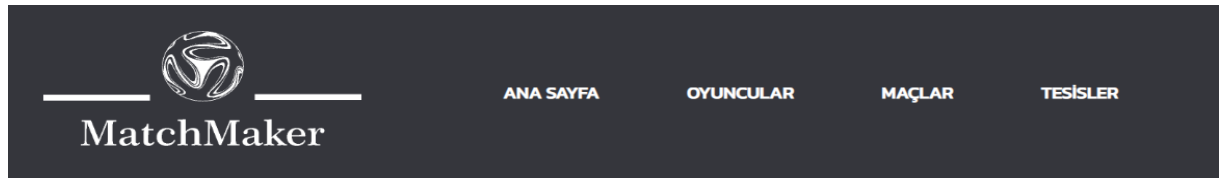


Figure 40.

Int. 13 Footer.



Figure 41.

3.5. RECOMMENDATIONS

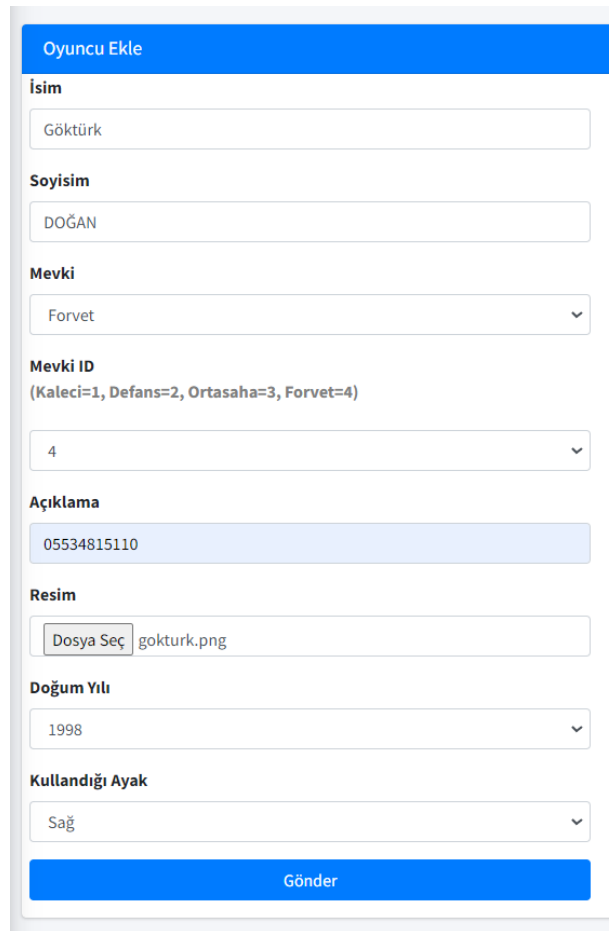
The project requires a PhpMyadmin database structure to be formed manually for the software. Use of Php programming language may be beneficial for the ease of implementation.

4. TEST PLAN AND TEST REPORT

To test the matchmaker project, I included a few of the football organizations that I organized myself, which was the reason for the idea of the project.

4.1. ADDING PLAYERS

First of all, I entered the information of the players who are likely to take part in the matches into the system.



Oyuncu Ekle

İsim
Göktürk

Soyisim
DOĞAN

Mevki
Forvet

Mevki ID
(Kaleci=1, Defans=2, Ortasaha=3, Forvet=4)
4

Açıklama
05534815110

Resim
Dosya Seç gokturk.png

Doğum Yılı
1998

Kullandığı Ayak
Sağ

Gönder

Figure 42.

I checked whether the players I added were successfully registered to the system and took a look at the pool of players.

Oyuncular Listesi

Show 10 entries

Search:

#	Adı	Soyadı	Mevkisi	Kullandığı Ayak	Yaşı	İşlem
1	Göktürk	DOĞAN	Forvet	Sağ	24	Düzenle Kaldır
2	Oğuzhan	ŞENOKUR	Kaleci	Sol	24	Düzenle Kaldır
3	Burak	ÖZER	Defans	Sağ	24	Düzenle Kaldır
4	Mustafa	ÖZKAN	Defans	Sol	25	Düzenle Kaldır
5	Melih	ERTÜRK	Ortasaha	Sağ	22	Düzenle Kaldır
6	Hakan	KESKİN	Defans	Sağ	23	Düzenle Kaldır
7	Uğur	ONAR	Forvet	Sağ	23	Düzenle Kaldır
8	Halil	REİSOĞLU	Ortasaha	Sağ	23	Düzenle Kaldır
9	Doğancan	MAVİDENİZ	Ortasaha	Sağ	22	Düzenle Kaldır
10	Giray	UÇAR	Forvet	Sağ	22	Düzenle Kaldır
#	Adı	Soyadı	Mevkisi	Kullandığı Ayak	Yaşı	İşlem

Showing 1 to 10 of 40 entries

[Previous](#) [1](#) [2](#) [3](#) [4](#) [Next](#)

Figure 43.

4.2 ADDING FACILITY

I checked whether the facilities that provide football services in nearby locations were entered into the system and added to the table.

Tesis Ekle

Tesis Adı

Çamlık

Adres

Çamlık, Cahit Zarifoğlu Cd. No:7, 34912 Pendik/İstanbul

Saha Tipi

Kapalı

Telefon Numarası

05333012072

Resim

Dosya Seç Dosya seçilmedi

Saha Sayısı

2

Konum

https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1

Gönder

Figure 44.

Tesisler Listesi						
Show 10 entries		Search:				
#	Tesis Adı	Saha Sayısı	Telefon	Saha Tipi	Adres	İşlem
1	AydınOral	3	05333012072	Kapalı	Çamlık, Cahit Zarifoğlu Cd. No:7, 34912 Pendik/İstanbul	Düzenle Kaldır
2	Çamlık	3	(0216)5865775	Kapalı	Çamlık, Erguvan Sokağı No:2, 34912 Pendik/İstanbul	Düzenle Kaldır
3	Arena	2	05300611717	Kapalı	Harmandere, Ankara Cd., 34912 Kurna/Pendik/İstanbul	Düzenle Kaldır
4	İrem	1	(0216)4826061	Açık	Yenişehir, Mustafa Akyol Sokağı, 34912 Pendik/İstanbul	Düzenle Kaldır
#	Adı	Soyadı	Mevkisi	Kullandığı Ayak	Doğum Tarihi	İşlem
Showing 1 to 4 of 4 entries						
						Previous 1 Next

Figure 45.

4.3 ADDING MATCH

I added a match to my site and created two separate teams from the players added to the pool.

Maç Ekle

Tarih

18.05.2022

Saat

15:30

Tesis

Çamlık

Gönder

Figure 46.



Figure 47.

After the match was completed, I made data entries from the management panel to give the scores of the players and to make the entries of the scores produced by the teams.



Figure 48.

Since all the features in the system work smoothly, no problems were encountered during the testing phases.

5. DISCUSSION AND CONCLUSION

5.1 Successful Aspects

The Matchmaker project has succeeded in solving two basic problems that were set out in line with the purpose of its development. A player pool has been created for the players and the matches have been selected by the administrator with the help of select inputs. Players are given a point by the manager after the matches are over.

5.2 Seficiencies

The most basic shortcoming of this project in my eyes is that the scores given to the players by the manager after the matches are not kept in an external table. This event brings out the following result, the scores of the players are not calculated in one place and recorded in their tables as the average score, so before adding players to the match, the manager can be content with searching for the player from the search button and looking at the points he has received in the past matches.

5.3 Suggestions

In order to make the project more useful, a membership opening panel can be added to the system, so that a new user profile can be created as anonymous user, member users other than administrative users. By taking advantage of the functions brought by this profile, a comment system can be added after the matches, in which the member users indicate their votes for the relevant match. Thanks to this system, the objectivity of the scores given by the administrator user to the players can be discussed and an interactive structure can be added to the system.

SOURCE

- <https://www.udemy.com/>
- <https://www.youtube.com/c/PhpT%C3%BCrkiye>
- <https://adminlte.io/>
- <https://bootstrapmade.com/>

GOKTURK DOGAN

Millet Caddesi, B Blok 14, Pendik, İstanbul 00034 | 05534815110 | gokturk.dogan@hotmail.com

Professional Summary

- Hardworking Software Development focused on completing work quickly to consistently exceed targets. Reliable team member accustomed to taking on challenging tasks. Dedicated to business success.

Skills

- Software Design
- Team Collaboration
- Object-Oriented Programming
- JS Frameworks: React, Angular, Vue
- Project Management
- RDMS Development and SQL
- Responsive Web Design
- Front-End Skills: HTML, CSS
- Front-End Programming: HTML5, CSS3, AJAX
- RESTful APIs
- CSS
- Project Updates
- Code Writing
- Data Structures and Algorithms
- Programming Languages: Java, C#, .Net, Python
- Source and Version Control: Git, Github
- Full Software Development Lifecycle
- Javascript Libraries and Frameworks
- Scripting Languages: PHP, Python, Javascript
- Back-End and Database Skills
- JavaScript

Experience

Software Developer Intern 06/2022 to Current
IMB Robotics – Istanbul, Pendik

- Worked closely with development team to implement coding changes and troubleshoot bugs.
- Used JavaScript, ASP and C++ to plan, test and implement various web-based applications.
- Designed and updated software systems to meet customer needs and exceed expectations.

Education

Bachelor of Science: Computer Engineering Jun 2023
Okan University - Istanbul , Tuzla

- Student Scholar Athlete
- Volunteer, [HSD Campus].
- Member, [Computer Science].

High School Diploma: Numerical Jun 2017
Kırımlı Fazilet Olcay Anadolu Lisesi - Istanbul , Pendik

- Graduated with honors.
- Student Government.
- Student Scholar Athlete

Languages

English, Bilingual
Okan University C Level Prep Class