

# **CENG 492 PROGRESS REPORT – 7**

**Group Name:** iTeam4

**Group Members:**

- Emilbek Joldoshev 1592476
- Hassan Salehe Matar 1591114
- Mehmet Barış Özkan 1560747
- Hüseyin Lutin 1560408

**Project Name:** Online National Election Voting System

In this document the weekly activities done by the iTeam4 for the senior project are described and the progress of the project is stated.

### **Mehmet Barış Özkan**

This week we have worked on the design of the general election voting pages that will be used by voters during the voting process. In the first version of our pages we have designed the pages with big party emblems and candidate pictures. And candidate list was coming after the party list. We have decided to change this design with two list namely party and independent candidate list near to each other and with small images. Each of the table has the value of party order, name, emblem and a selection radio button. At the end we come with such an interface;

2011 GENEL SEÇİMLERİ OY VERME SAYFASI

UYARI: Oy Vermek İstediginiz Aday ıa da Partisi Mousse Yardımıyla Seçiniz

SIRA	PARTİ ADI	AMBLEM	SEÇENEK
1.	Adalet Ve Kalkınma Partisi		<input type="radio"/>
2.	Demokrat Parti		<input type="radio"/>
3.	Eğilim Ve Demokrasi Partisi		<input type="radio"/>
4.	Cumhuriyet Halk Partisi		<input type="radio"/>
5.	Emek Partisi		<input type="radio"/>
6.	Millet Partisi		<input type="radio"/>
7.	Liberal Demokrat Parti		<input type="radio"/>

SIRA	ADAYIN İSMİ	RESİM	SEÇENEK
1.	Mehmet Barış Özkan		<input type="radio"/>
2.	Almet Coşkun		<input type="radio"/>
3.	Ali Üstün		<input type="radio"/>
4.	Murat Hakan		<input type="radio"/>
5.	Mustafa Yılmaz		<input type="radio"/>
6.	Hüseyin Lutin		<input type="radio"/>
7.	Sedat Tekin		<input type="radio"/>

At the end of the page there is also a submit button to save the voter's choice in the system.

Since we have started to develop our project in .NET environment this week we have also done some applications to get more familiar with the C# language and Visual Studio.

### ***Hassan Salehe Matar***

Apart from other assignments of the project we have a duty to migrate from Java to .NET. Therefore, we converted what we had developed in java on the Station Supervisor functionalities to .NET and tested them. The complete reference to the codes corresponding to the Stations supervisor program is as follows:

1. Master Page:

\* `svn+ssh://e1591114@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Areas/eMode/Views/Shared/StationMaster.Master`

2. Login page:

-view: `svn+ssh://e1591114@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Areas/eMode/Views/Supervis/Login.aspx`

3. Check-in

-view:

`svn+ssh://e1591114@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Areas/eMode/Views/Supervis/electOptions.aspx`

4. Offline votes

-view: `svn+ssh://e1591114@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Areas/eMode/Views/Supervis/Offvotes.aspx`

5. Poll Results

-view:

`svn+ssh://e1591114@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Areas/eMode/Views/Supervis/PollResults.aspx`

6. Controller

`svn+ssh://e1591114@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Areas/eMode/Controllers/SupervisController.cs`

### ***Emilbek Joldosev***

This week we developed our project in .Net.

We use MS SQL Server as a Database Management System.

To get data from and to database we use LINQ TO SQL<sup>1</sup>. LINQ TO SQL is a Mapping Tool of Relational Data to Object Model.

ASP.NET MVC<sup>2</sup> framework is our main tool in our application development. It separates Model, View and Controller with each other. And it has built-in test tool for testing application.

We designed two different styles for voter user interface. Tomorrow we will show and choose better one together.

We have completed voter login control and user authorization, authentication parts.

Tomorrow, we will try to finish voting part till our meeting.

In application, user authentication and authorization are done with a class called Authentication.cs<sup>3</sup>.

`SessionControlFilter` is used to control if the user is logged in or not. If it not logged in, then it will redirect user to Login Page.

`AuthenticationFilter` is used to control if the user is logged in AND HAS AUTHORIZED to that page. It is mostly useful when administrators, candidates are only allowed for specific pages.

Another main class is User class<sup>4</sup>.

In order to make abstraction over user we use this class.

It has Login, and getUser Methods:

`Login` method tries to login with given parameters, if successful then stores the user to session and returns true. If it is not successful returns error value.

`getUser` class returns current user object.

[1] <http://msdn.microsoft.com/en-us/library/bb425822.aspx>

[2] <http://www.asp.net/mvc>

[3] `svn+ssh://e1592476@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Models/Authentication.cs`

[4] `svn+ssh://e1592476@external.ceng.metu.edu.tr/depo/svn/490.2010/iteam4/ONEVS/Areas/eMode/Models/User.cs`

### ***Hüseyin Lutin***

I tried to make "Normal Interaction Mode user interface sequence design" but I am not familiar with C# I did not complete but I can show the code that I have written.

Since yesterday I have an exam at the meeting time I will not be able to attend the meeting. Therefore, I will show you next week

thanks