**Başar Çelebi - 20201701066**

**Şafak Gün - 20201701001**

**Doğukan Gökalp - 20201701031**

**Öykü Sucuoğlu - 20201701057 Project Report#2**

**Project Report**

We wanted to create a voting system where user entries are partially made with username and password entries. In our voting system, the user can change their password, can vote, and our program will store the votes of the users until the voting process ends. Even if the user exits the program, the folders we create will store the information. When we start our program, the first thing the user will see will be the login section. He gets the chance to vote by entering his username followed by his password to continue. If the password entered by the user is invalid, the user will have unlimited attempts to enter the correct password to continue. We also created an option for the user to change their password if they want. When a user logs into the voting system, a folder will appear that stores the information that our administrator has written in order. When the user wants to change the password, a screen will appear asking him to type his new password. However, he will be asked to type this password 2 times. Thus, we will test whether the user confirms the newly entered password. If the new password entered for the first time and the password entered for the second time do not match, a warning will be sent to the user. Thus, the process will continue until the user types both passwords correctly. After changing the password, the new login information will be saved.

After logging in, our program will present the user with an option on whether he wants to vote or not, if he enters a valid number, he can vote. If he did not enter the correct number, our program will say, “You have entered an invalid number. Please log in again". The user will choose what they want to vote for: Yes, No, Abstain. After each vote, a sentence appears: “Your vote has been saved as '...'”.

After voting, the user will also have the opportunity to change the vote again. We also created folders to store users' votes to count them. We created a folder named vote\_of\_people to see each user's vote in turn, and if they change the vote, the vote will also change from that folder. We based our program on 4 users but also created a function for new users to log in. Abstention votes were excluded from the total number of users as they did not affect the total count to decide whether to win. Thus, we made it easier to calculate the result by subtracting the votes of the users who abstained from the number of people. Additionally, if three of the users vote the same (yes or no), the voting will end as the vote of the other admin is not important. In addition, the numbers of yes, no and abstention votes are also stored in different folders. Thus, even if we close the program, the voting process will continue from where it left off when the user restarts the system, as the vote numbers and votes are recorded.

As a result, our program successfully processed and stored every vote count. A program has been created where we can call the available data whenever we need it. We have developed a program that saves users' voting, changes passwords, and records all processes when the user leaves before completing the transaction.