



ÖZYEĞİN UNIVERSITY  
FACULTY OF ENGINEERING  
DEPARTMENT OF COMPUTER SCIENCE

**CS 402**

**2018 Summer**

**SENIOR PROJECT REPORT**

**Online Recruitment Application**

By  
**İbrahim Enes ALAN**

Supervised By  
**Assoc. Prof. Erhan ÖZTOP**

**Declaration of Own Work Statement/ (Plagiarism Statement)**

Hereby I confirm that all this work is original and my own. I have clearly referenced/listed all sources as appropriate and given the sources of all pictures, data etc. that are not my own. I have not made any use of the essay(s) or other work of any other student(s) either past or present, at this or any other educational institution. I also declare that this project has not previously been submitted for assessment in any other course, degree or qualification at this or any other educational institution.

**Student Name and**

**Surname:**

**Signature:**

**Place, Date:**

# Abstract

*This report addresses the problem of hiring new employee and this process. My solution approach is creating new online recruitment application. This application aims to shorten the recruitment process. From my observations I can easily say Human Resources Departments has massive workload in this hiring process. The application has too many capabilities. Most important feature is system can be able to take relevant data on candidate's CV and store it in database. Thanks to this feature candidate's workload will be minimized. Also, by using this application admin can easily list the candidates and give success point to them. Also, admin can arrange job advertisements. On candidate side of the system, candidates can list and apply a job. Also, they can track hiring process. Another important feature is candidates have their personal profiles which can be updated at any time. In this application some pages needs to be updated constantly. To provide this feature dynamic pages created with using AJAX. The application is web-based, and It will be further developed.*

# Contents

<b>I. INTRODUCTION .....</b>	<b>4</b>
<b>II. BACKGROUND.....</b>	<b>6</b>
<b>III. PROBLEM STATEMENT.....</b>	<b>8</b>
<b>IV. SOLUTION APPROACH .....</b>	<b>14</b>
<b>V. RESULTS AND DISCUSSION .....</b>	<b>26</b>
<b>VI. RELATED WORK.....</b>	<b>31</b>
<b>VII. CONCLUSION AND FUTURE WORK.....</b>	<b>34</b>
<b>ACKNOWLEDGEMENTS.....</b>	<b>37</b>
<b>REFERENCES.....</b>	<b>38</b>
<b>APPENDIX.....</b>	<b>39</b>

## **I. Introduction**

Recruitment process is for hiring new employees. This process can be stressful for both company and candidate sides. When we look at the candidate side, work is the most important part of our live, we need to enhance ourselves by doing this also we need to make money and continue our lives. According to human resources departments, an average candidate waits fifteen-twenty days on this hiring process. [1] Also, candidates apply to many jobs at the same time. Most of them cannot wait too long process time.

When we look at company side, hiring process has too many steps. First one is to identify what they need. In this part Human Resources department specify what capabilities they want from the new employee. After this, company creates recruitment advertisement and pass to the search step. This step can be really long. Because there are too many options. Most of them are not capable for specified job. Also, companies do this process with using different platforms. For example, Kariyer.net [2] or LinkedIn [3]. Human Resources needs to compare all candidates which increase their workload because they are in different platforms. My solution is every company needs recruitment process applications and they would like to easily view and compare candidates by using their platforms.

My project is to build Online Recruitment Application which solves the two main problems. In the search step both the employee and company lose enormous time. Sometimes this causes, the candidate to start another job or company transfers workload to someone who is in same department in company. Second problem is Human Resources workload. When this project starts to use in companies, HR's can easily filter, and search candidates based on which criteria they want.

Another problem solved by using this application which is candidates must fill too many personal information areas. For example, personal information, education, experience, skills. As I mentioned before candidate can be tired while filling all these labels. In my application candidates can upload their CV's pdf or image format and by using this CV, Java extension is able to fill most of the labels. This feature decreases candidates' workload. This fill function has three main steps on the back-end side. First one is Object Character Recognition. Second one is detecting keywords and Separate valuable information. And the last part is storing this information to database's relevant fields.

The system is a Web-based application; therefore, I am using HTML and CSS to design the website and make it functional. For the back-end development, I am using MySQL, Java and PHP. For Optical Character Recognition and Pattern Recognition implementation I am using Java, the reason I chose Java to develop the engine was that there are libraries that suited my tasks in Java.

This report is a detailed explanation of Online Recruitment Application project. The set of tools, techniques and technologies that are used in this project can be found in the following section. Third Section introduces the project scope, the engineering problems that I solved, any assumptions and constraints that were faced while developing this project, and all the functional and non-functional requirements of the project. In the Section 4, the project is described in detail, with the tools and techniques that were used, and technical, operational and financial feasibility of our implementation. The knowledge and skill set that were used in this project and the engineering standards that were taken into consideration in the development process is also introduced in this section. Section 5 discusses the results that are achieved so far and the status of the project, as well as the limitations I have faced. Section 6 explains the related works and their comparison with our project. Seventh Section gives a summary of my achievements so far, expected social and economic impacts of our work, as well as its effect on global and business and environment. The safety and security issues and legal consequences are discussed in this section. This section also provides an insight of our future work and how the project can be extended.

## II. Background

In the project mostly, I use PHP, SQL and Java. For the design I use Scripting Languages which are CSS and JavaScript. My main computer's operating system is Windows 10. Web Server's operating system was CentOS, but I changed after this to Ubuntu version 16.04.

**MySQL Workbench:** MySQL Workbench was used to create and manage the database using SQL queries. A database holds the necessary information that can later be used or updated. In our case, the database holds the information about candidates, admins and candidate related CV's.

**Wamp Server:** Basically, Wamp stands for Windows Apache MySQL and PHP. In this project Wamp provided, testing of web-pages in a localhost. It runs local Apache Server. Also, by using this software, I can communicate with the database and test my webpages.

**Notepad++ V7.5:** I used Notepad++ to implement my HTML and PHP codes. I do not want to use workbenches like Dreamviewer. The reason I chose this software was that it is user-friendly and easy to use.

**Digital Ocean Server [4]:** I also purchase Digital Ocean Server which located in New York. This purchase free for students who has GitHub Education Pack. My droplet has 25 GB SSD Disk and 1TB Bandwidth which is used to control web traffic. Also, my website has IPv6 internet protocol. Operating system of the server is Ubuntu.

**Putty 0.70 [5]:** Putty is for Windows, to make connection with Unix-based platforms with using official ports. By using this application, I can connect my server remotely. In Putty SSH Key Generator is also included software which is called PuttyGen.

**Java:** Java is used to implement Separation Algorithm and Object Character Recognition.

**Eclipse Photon:** While working with Java I am using Eclipse Photon to implement my Java code. The reason I chose this IDE was that I am familiar with this and it is easy to use.

**FileZilla Client 3.33:** I use FileZilla to make File Transfer Operations between my computer and my located server. I choose this software because It has user friendly and easy to understand User Interface.

**Apache 2.0:** Apache's simple definition, we can say web server software. To handle php code I install both my Localhost and Server. When installing server which operating system is Ubuntu, I issued some problems with communication port however after some research I handled it.

**PHP Mailer 5.2 Stable Version [6]:** It is an enhanced mailer class. I use this because I need to inform candidate about Apply Job Information, with your application has been received message. Also, by using this class I can send HTML based e-mails.

**PHP-Java Bridge [7]:** By importing this project to my Java Project I open a port which infinitely listen for incoming PHP requests. I am using port 8080 to ensure communication between Java and Web Application.

**Tesseract OCR [8]:** Tesseract is Optical Character Recognition project which is started by Google. When we compare Tesseract to other OCR Projects it gives more accurate results. Tesseract written by using C++. I implement wrapper to use this package on my Java project.

**JsPDF [9]:** It is a library that generate PDF's on the client side of Application. I implement this in my project because admins want to export candidate ranking table or candidates' contact information. But JsPDF does not support UTF-8 Unicode Characters.

**jQuery [10]:** This technology used for creating dynamic web pages using Ajax also creating and set tasks to DOM elements will be easy with using jQuery JavaScript Library. I used jQuery version 3.3.1 which is last version.

### **III. Problem Statement**

There are two main problems at the recruitment process. Candidates lose enormous time while waiting job advertisement results. Second one is Human Resources department has difficulty comparing many candidates. I realize these problems while working at the Vodafone HR department. After this I take my supervisor's approval and pick this project topic.

- **Project Scope**

Online Recruitment Application is a Web-based Project which will shorten the hiring process. It is new project but, in the web, there are similar applications, but they have limited capabilities and it occurs some problems in hiring process. One of the relevant sample is Kariyer.Net [2], however it did not increase efficiency it creates more workload to the HR. My application will allow candidates show job announcements and apply them with using their profiles. So far it has some capabilities like Kariyer.Net, but in my project admins can easily group and filter users by using this filtering technique most of the candidates will be eliminated. After this Human Resources easily investigate and give points to the rest of the candidates by using this application. Application's most important feature is called auto fill CV function. Thanks to this feature candidate can fill most of the information labels by clicking only one button.

- **Engineering Problem**

When we consider recruitment process, main problem occurs investigating candidate's CV. Because there are a lot of position and too many candidates in sample company. This project's aim is develop a website which store candidate CV's online and Human Resources admin easily search and eliminate unnecessary information on CV's. This project has engineering problem by using this application companies can arrange their resources effectively. Also, candidates will be reach hiring result in a shorter time. Also implementing data separation and image recognition algorithms was big challenge for me. Another problem was to make connection between Java Application and Web Application.



- **Assumptions**

1. HR representatives are having hard time handling multiple job seeking environments.
2. User should upload CV before using auto fill function.
3. I assume Candidate CV's will be readable and clear format.

- **Constraints**

1. The application requires internet connection since it will be Web-based.
2. CV patterns may be change from candidate to candidate.
3. Some browsers web structure can be change and for Internet Explorer export table as Excel File function cannot work properly.

- **Functional and Non-functional Requirements**

Requirement's ID determined for candidate capabilities "CAN", for admin capabilities "ADM". If requirement is for both users' ID will be "ORA".

**Functional Requirement 1:**

**ID:** ORA-001

**Title:** User Profiles

**Description:** This software should have two types of user profiles which are Candidate and Admin.

**Functional Requirement 2:**

**ID:** ORA-002

**Title:** User Registration

**Description:** This software user should be able to register with using Web Site by entering username and password.

**Functional Requirement 3:**

**ID:** ORA-003

**Title:** Use Registration Unique Username

**Description:** When user wants to register new account, Username must be unique. If it is not this software should warn the user.

**Functional Requirement 4:**

**ID:** ORA-004

**Title:** User Login Incorrect Password

**Description:** This software should show error message to warn the user for incorrect password situation.

**Functional Requirement 5:**

**ID:** ORA-005

**Title:** Successful Login

**Description:** After successful login both admin and candidates will be directed their homepages.

**Functional Requirement 6:**

**ID:** ORA-006

**Title:** SQL Injection

**Description:** This software should handle SQL Injection while login to system.

**Functional Requirement 7:**

**ID:** ORA-007

**Title:** Preparing Dataset

**Description:** This software should prepare the necessary dataset.

**Functional Requirement 8:**

**ID:** ADM-001

**Title:** Candidate Information

**Description:** This software should show candidate information to the Company Admin.

**Functional Requirement 9:**

**ID:** ADM-002

**Title:** Adding A Candidate

**Description:** Admins should add new candidate by using this software.

**Functional Requirement 10:**

**ID:** ADM-003

**Title:** Add Candidate Language Option

**Description:** When filling new candidate information, software should create language options by looking language number label.

**Functional Requirement 11:**

**ID:** ADM-004

**Title:** Deleting A Candidate

**Description:** Admins should delete existing candidate by using this software.

**Functional Requirement 12:**

**ID:** ADM-005

**Title:** Update Acceptance Status

**Description:** Admin can update candidate acceptance status when candidate wants to be interviewed.

**Functional Requirement 13:**

**ID:** ADM-006

**Title:** Search Candidate

**Description:** Admins should search candidate with using keywords. This search candidate function has filter option.

**Functional Requirement 14:**

**ID:** ADM-007

**Title:** Show and Print Candidate Information

**Description:** After searching candidate admins should be able to show candidate information and profile picture. If desired admin can print this page.

**Functional Requirement 15:**

**ID:** ADM-008

**Title:** Add New Job Advertisement

**Description:** By using this software admin can add new job advertisement.

**Functional Requirement 16:**

**ID:** ADM-009

**Title:** Applied Job Information

**Description:** This software should list which job advertisement applied by candidates and these candidates' information to the Company Admin.

**Functional Requirement 17:**

**ID:** ADM-010

**Title:** Evaluate A Candidate

**Description:** Admins should give evaluation point to the candidate by using this software.

**Functional Requirement 18:**

**ID:** ADM-011

**Title:** Show CV Option

**Description:** Before giving the evaluation point to candidate, admin can check candidate CV's with using show pdf option

**Functional Requirement 19:**

**ID:** ADM-012

**Title:** Automatic Evaluation Point

**Description:** This software should create automatic evaluation point for every candidate.

**Functional Requirement 20:**

**ID:** ADM-013

**Title:** Automatic Evaluation Point Settings

**Description:** By using this software admin can change automatic evaluation point settings.

**Functional Requirement 21:**

**ID:** ADM-014

**Title:** Export Candidate Information as PDF format.

**Description:** By using this software admin can export candidate contact information as a PDF format.

**Functional Requirement 22:**

**ID:** CAN-001

**Title:** Job Description View

**Description:** Candidates should view a job and job's description by using this software.

**Functional Requirement 23:**

**ID:** CAN-002

**Title:** Apply For A Job

**Description:** Candidates should apply a specific job by using this software.

**Functional Requirement 24:**

**ID:** CAN-003

**Title:** Apply Job Information Mail

**Description:** This software should mail, application received message to related candidate after candidate applies new job advertisement.

**Functional Requirement 25:**

**ID:** CAN-004

**Title:** Upload CV

**Description:** Candidates can upload their CV's to the Server by choosing file destination.

**Functional Requirement 26:**

**ID:** CAN-005

**Title:** Upload CV Document Type

**Description:** While uploading a CV file type must be pdf or image format. Otherwise software should warn the user about file extension.

**Functional Requirement 27:**

**ID:** CAN-006

**Title:** Upload CV Delete Older Versions

**Description:** While uploading a CV file software checks older versions if there is one, software needs to update with the newest one.

**Functional Requirement 28:**

**ID:** CAN-007

**Title:** Auto Fill Candidate Information using CV

**Description:** After upload CV, candidate can use auto-fill function. This function should be able to fill candidate information and related database fields automatically.

**Functional Requirement 29:**

**ID:** CAN-008

**Title:** Upload CV By Taking Image

**Description:** Candidates can upload their CV's to the Server by taking picture of it, with using webcam.

**Functional Requirement 30:**

**ID:** CAN-009

**Title:** Candidate Settings

**Description:** By using this software candidates can update their personal information.

**Functional Requirement 31:**

**ID:** CAN-010

**Title:** Candidate Profile Picture

**Description:** By using this software candidates can take profile picture and add this their account.

**Functional Requirement 32:**

**ID:** CAN-011

**Title:** Candidate Acceptance Status

**Description:** By using this software candidates can track their job applications.

**Functional Requirement 33:**

**ID:** CAN-012

**Title:** Candidate Status Information Mail

**Description:** This software should mail, when candidate acceptance status set to “Accepted” and should send interview information to candidate.

**Non-Functional Requirement 1:**

**ID:** ORA-008

**Title:** Response Time

**Description:** Web page response time is minimized according to the size of the data.

**Non-Functional Requirement 2:**

**ID:** ORA-009

**Title:** Connection

**Description:** This software has PHP-MySQL, Java-PHP and MySQL-Java connections.

**Non-Functional Requirement 3:**

**ID:** ORA-010

**Title:** Exception

**Description:** While connecting PHP and MySQL, software can handle exceptions and gives specific error message.

## **IV. Solution Approach**

At the beginning of the project I start with front-end side. In the project, I used object-oriented programming technique. Firstly, I choose to work with Agile Software Development for my project. If new feature needed in my project, I simply plan which part needs to be added to Site and start to work with this feature. During the installation of tools, I encountered with database port problem which wanted port currently using. To solve this issue, I update my database connection file. Another issue is related with Server provider which is Digital Ocean. My account is flagged and locked for abuse, I opened my account three day before this event. After creating support ticket, I solve this situation. And I need to install again Ubuntu, Apache, MySQL, PhpMyAdmin to the server.

Online Recruitment Application there are two types of candidate evaluation point. First one is given by the admin by looking candidate CV and motivation. Second one is created automatically with using basic algorithm. Candidate evaluation point ranking system takes candidate's some of information. These information are driving license, marital status, foreign language number, master education, disability condition and age. In the system if our candidate age is below 25 system assign this candidate as a young talent and candidate evaluation point will be increase with using young talent point which can be define by admin with using point settings section. These fields are determinative for admin.

Solution Approach should be divided into two sections as front-end and back-end sites.

### **Part 1: Front-end Development**

This part includes all HTML and CSS files. To get started easily implementing functions, I create prototype of the web page style. Simple web page with one web-page title, menu bar and one-page content part. Every section of the website has different HTML file. When user enter the site, user has two login options which are Login as Admin and Login as Candidate. By looking this login type menu bar changes because they have different capabilities.

In this project I use CSS for arranging presentation of HTML files. They stored in server in the CSS folder. There are 2 css files in this folder. First one is 'style.css'. By using this I handle container size, font family, background color, menu bar, hover settings, different margin specifications, different button types; size, color, position. And it includes too much main design contents. Second one is "takeimg.css". By using this I arrange candidate profile picture's resolution and size.

Also, in the project I use JavaScript to create necessary labels when they want. On add new candidate, update candidate functions foreign language number, experience and professional skills labels will be created by choosing number. When our selection number was changed it dynamically changes our created label number.

## Part 2: Back-end Development

This part includes all Web Site Structure, Database Design, OCR implementation, Auto Fill Function Design and explanation.

### • Web Site Structure

This project's most of the server-side functions handle with using Php. While connecting Web Site to the database I used Pdo [11]. I chose this connection technique because I am more familiar Pdo instead of Mysqli. And for me Pdo is more capable than Mysqli. Code for connection is stored file which name is 'databaseConnection.php'. It includes hostname, user name, password, database name besides these it also includes port number to handle with used port status. When my page needs to take information from database I call this file for the connect easily.

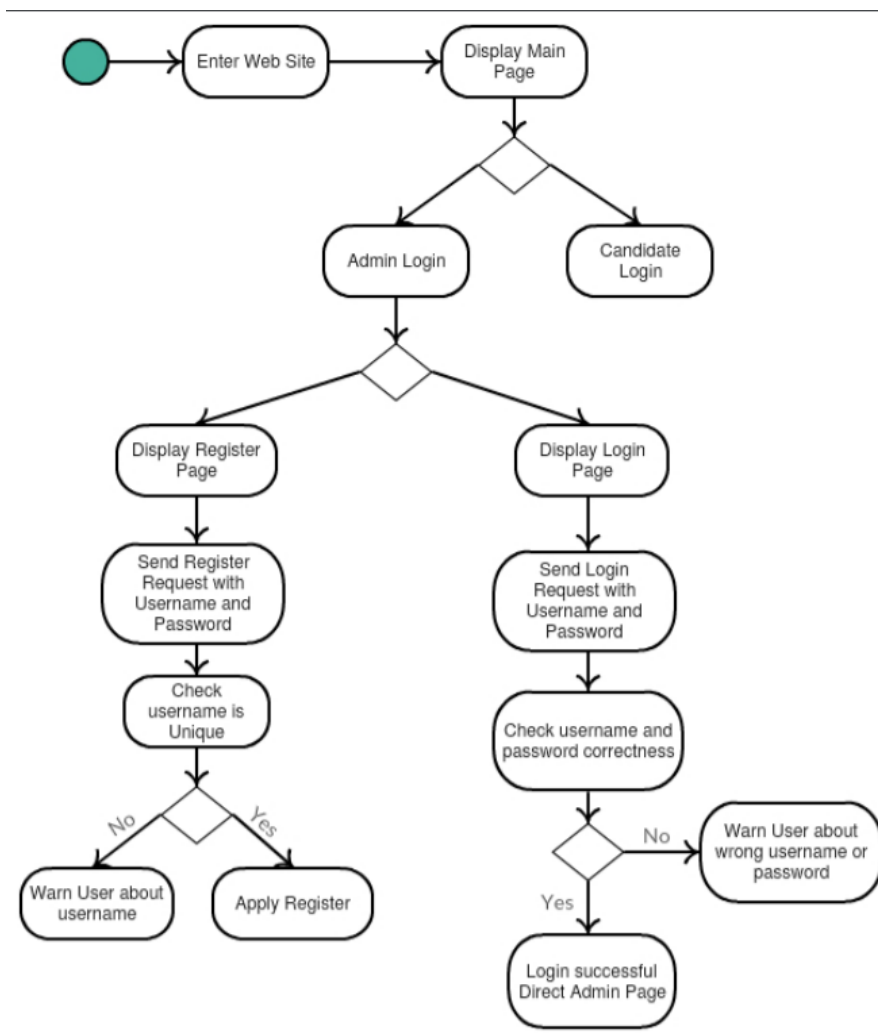


Figure [1]: Admin Login and Register System Activity Diagram

This application prevents undesirable logins and breaches with controlling session situation on all pages. When user login to the system, session will be started, and user name and password stored in this session. By using this technique, I can reach user name and password anytime. When user log out to the system, session will be destroyed, and stored user information is deleted.

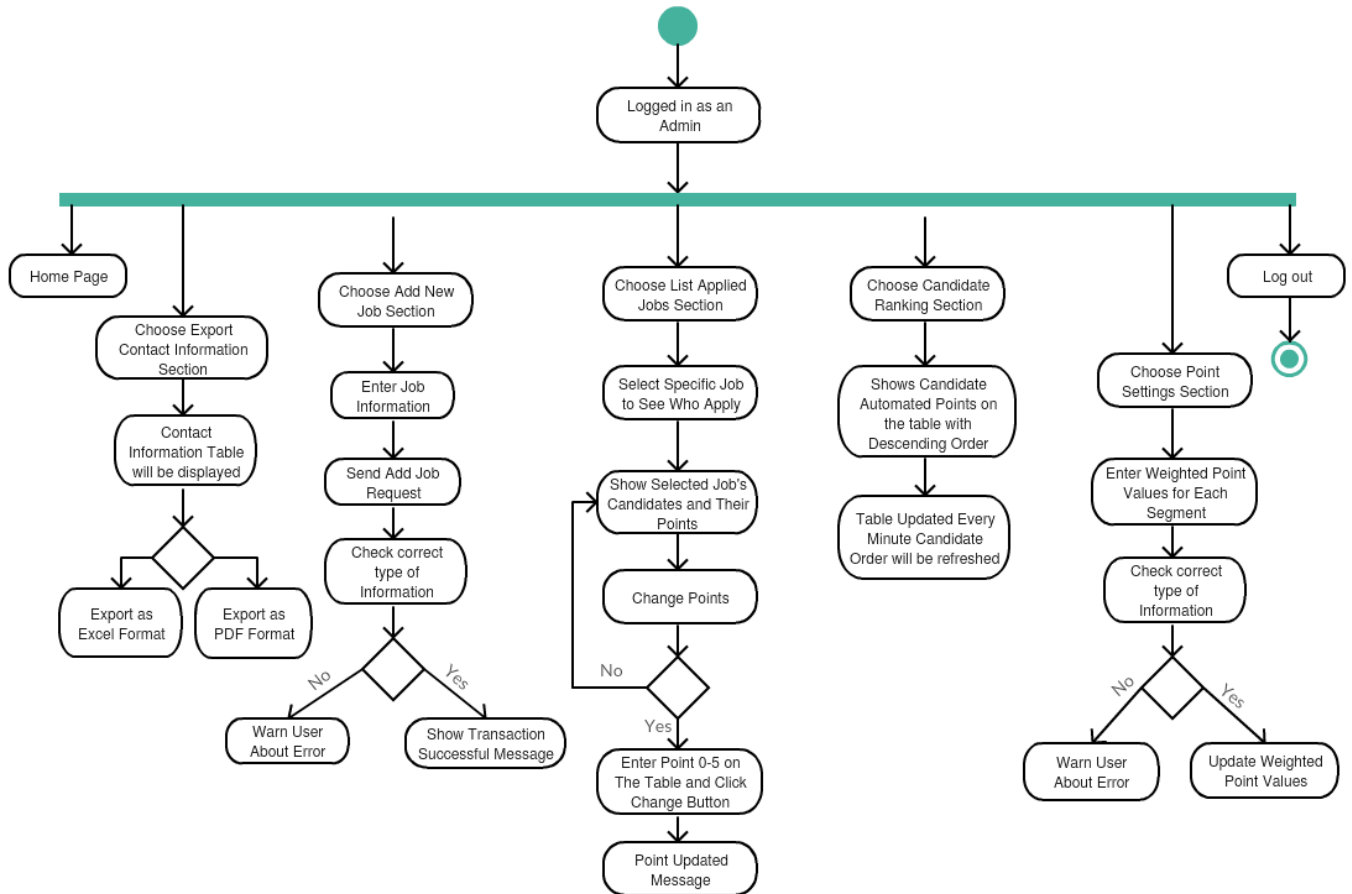


Figure [2]: Admin Panel's Activity Diagram

To increase readability of the diagram and not break the report's page structure. I divide candidate settings section and search candidate information section to another diagram.

By using evaluation scoring part in the software which will allow employer to get a first impression about candidate before interview.



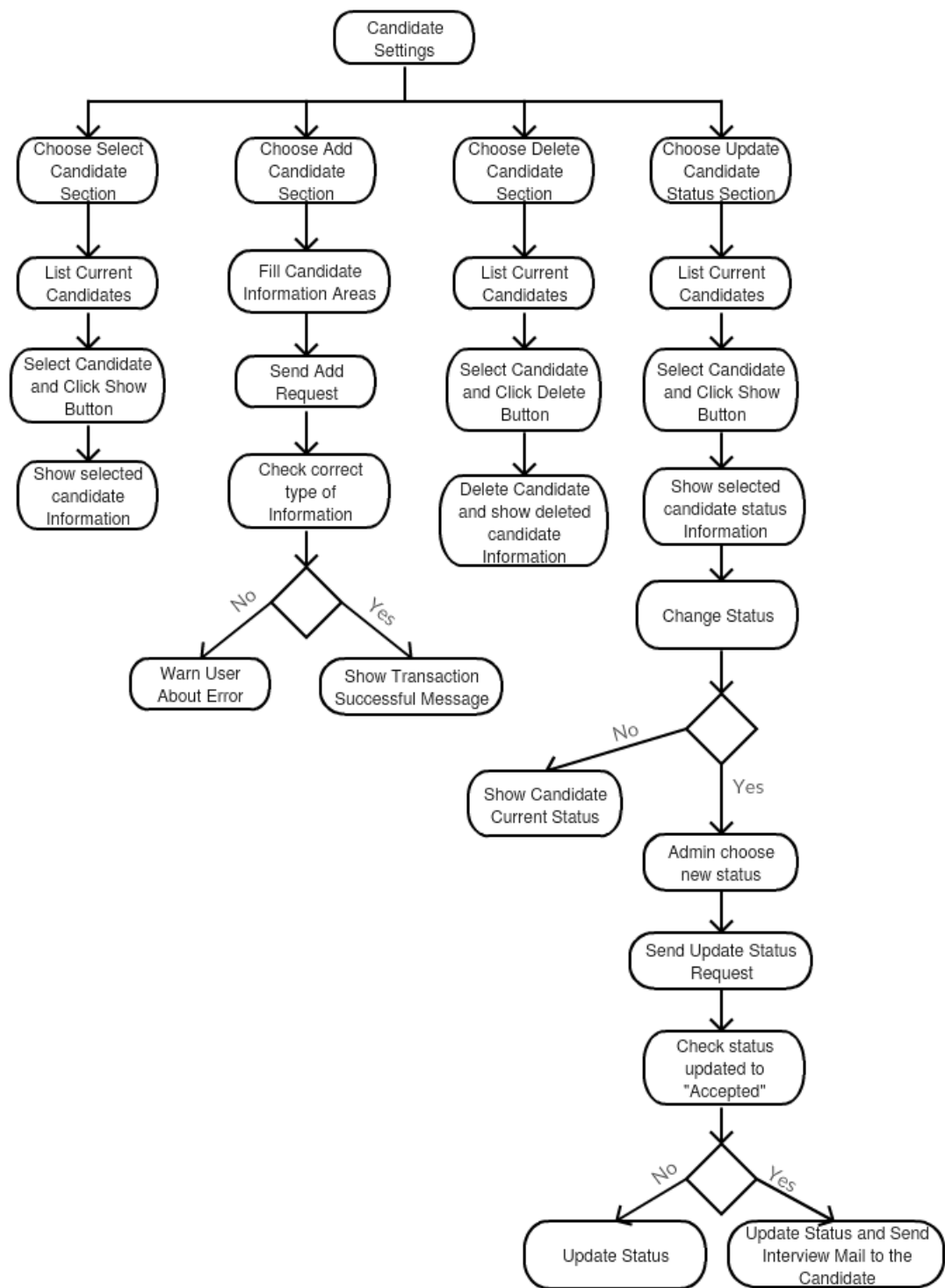


Figure [3]: Admin Panel's Candidate Setting Activity Diagram

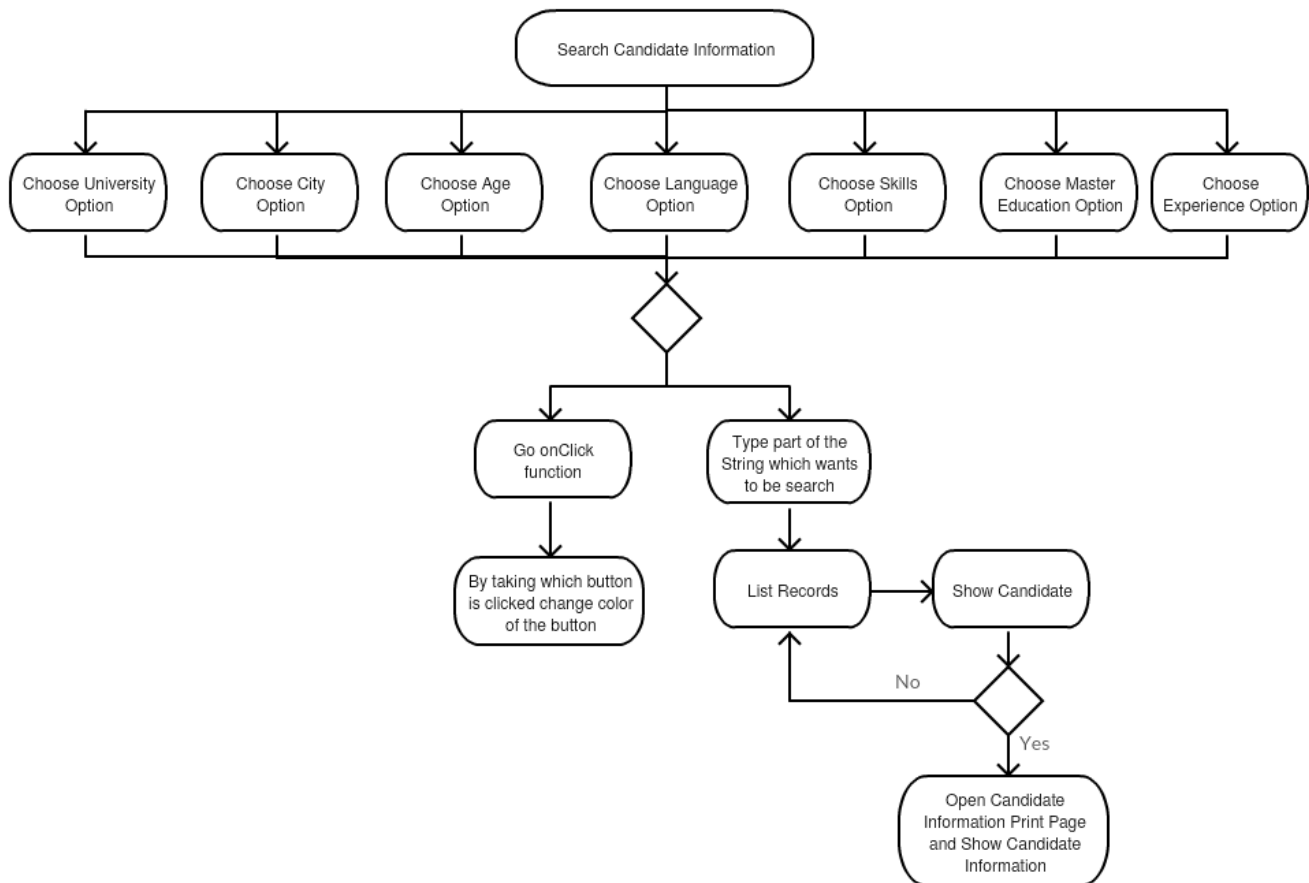


Figure [4]: Admin Panel's Search Candidate Information Feature Activity Diagram

Search Candidate Information feature works with 3 main JavaScript functions. First one is for change css file when button is clicked button colour will be change. [Figure 70] Another function which name is getStates, takes value in the input box when onkeyup function is triggered. Also, this function takes which button is clicked last as a parameter. Send these values to the related php file. On this file, query selection made by using button value. After this by using written String query will be executed. Until another button is clicked this option can be used for Live Search. Last function is for create hidden form and use show candidate feature. By using this we pass candidate variable's to showcandidate.php file. It creates new pop-up window which includes candidate information and print option. When Admin wants to see searched candidate information with details, it is enough to click show candidate button.

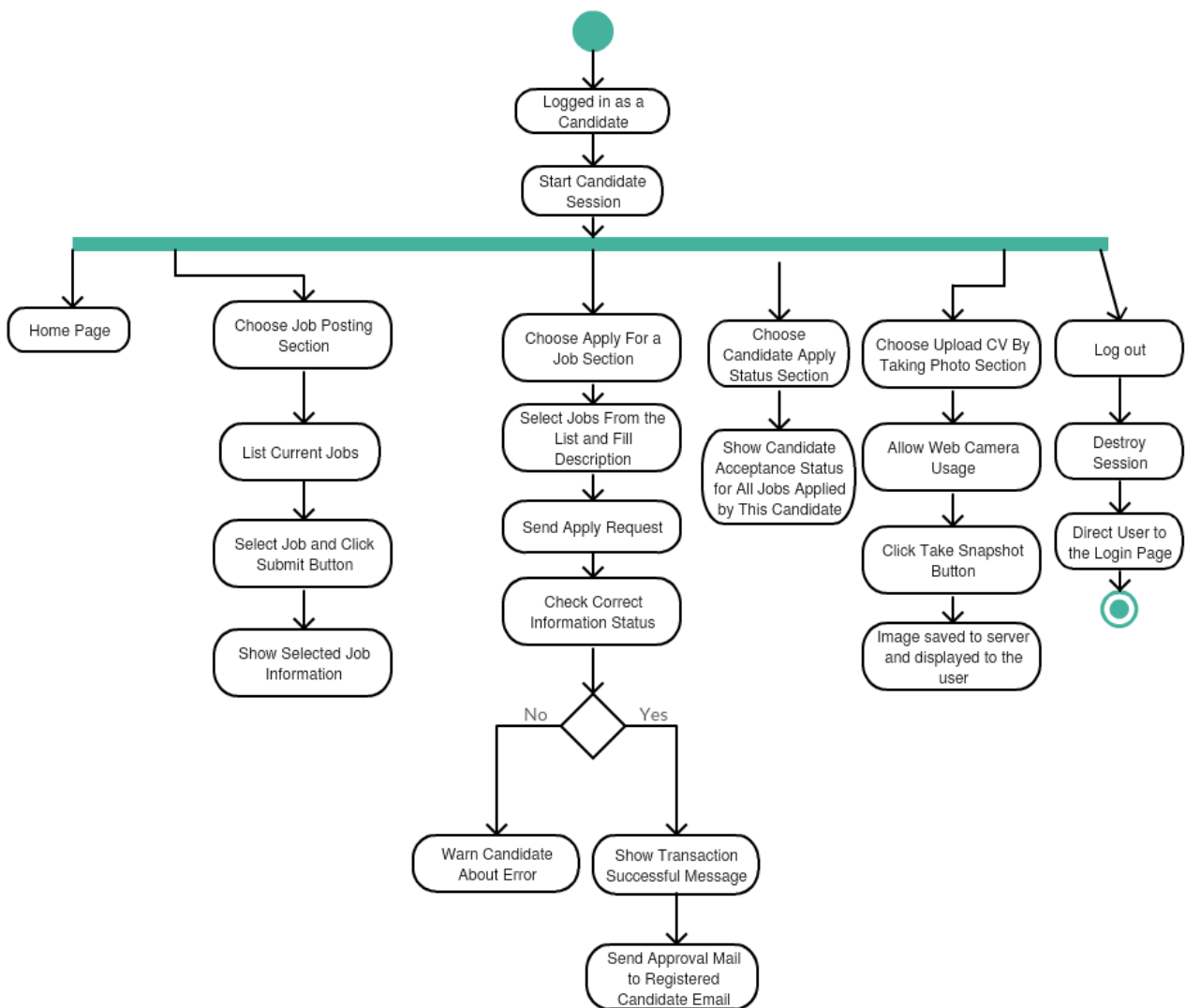


Figure [5]: Candidate Panel's Activity Diagram

To increase readability of the diagram and not break the report's page structure. I will explain, upload CV and Candidate Settings sections on another diagram. Apply for a job section is designed as a dynamic web page. Job selection menu is refreshed every 45 seconds. Thanks to this functionality when admin add new job advertisement on candidate side job selection menu will be updated. While implementing Upload CV by taking photo feature, first I store images on the database with blob type. But I cause this feature run slowly. To decrease response time of this feature I add this pictures on the specific folder in located Server. Database just keeps these images location and candidate information.

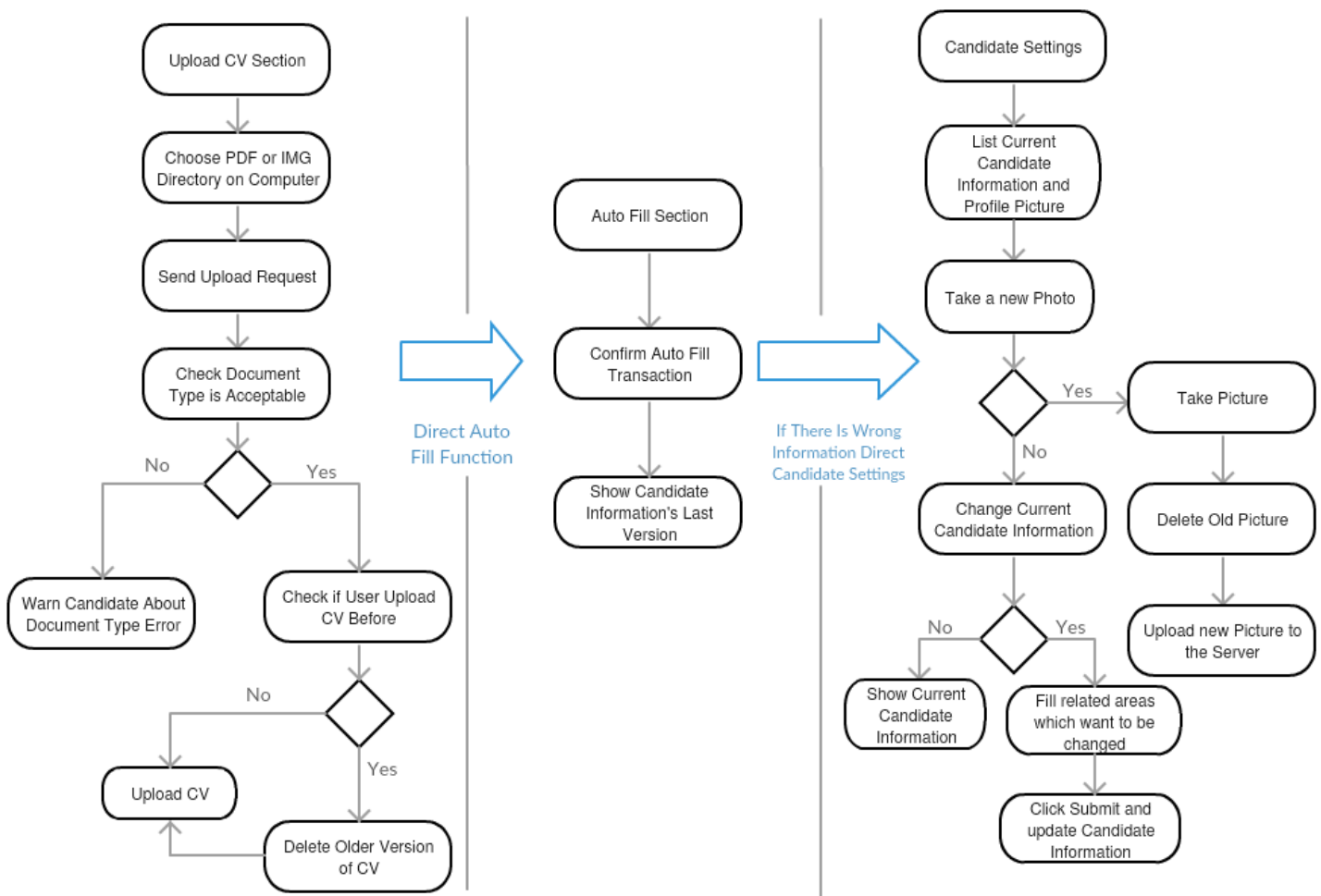


Figure [6]: Relationship between Upload CV - Auto Fill - Candidate Settings

Upload CV section there is two main condition. First one is checking document type. Application only allows pictures and pdf files. Second condition is check candidate upload CV before. When CV successfully uploaded candidate directs to the auto fill function confirmation page. If candidate confirms this transaction. Separation Engine will start. First uploaded document structure is converted to machine readable text with using OCR. This saves an output on the folder which name is same as uploaded pdf's name, but it is txt format. Program sends this output name and candidate Id as a parameter to folder read class. After searching relevant keywords and save them to the database, auto fill function is done. At the end we show last version of candidate information to the user.

- **Database Model**

You can see entity relation table diagram to understand my database model.

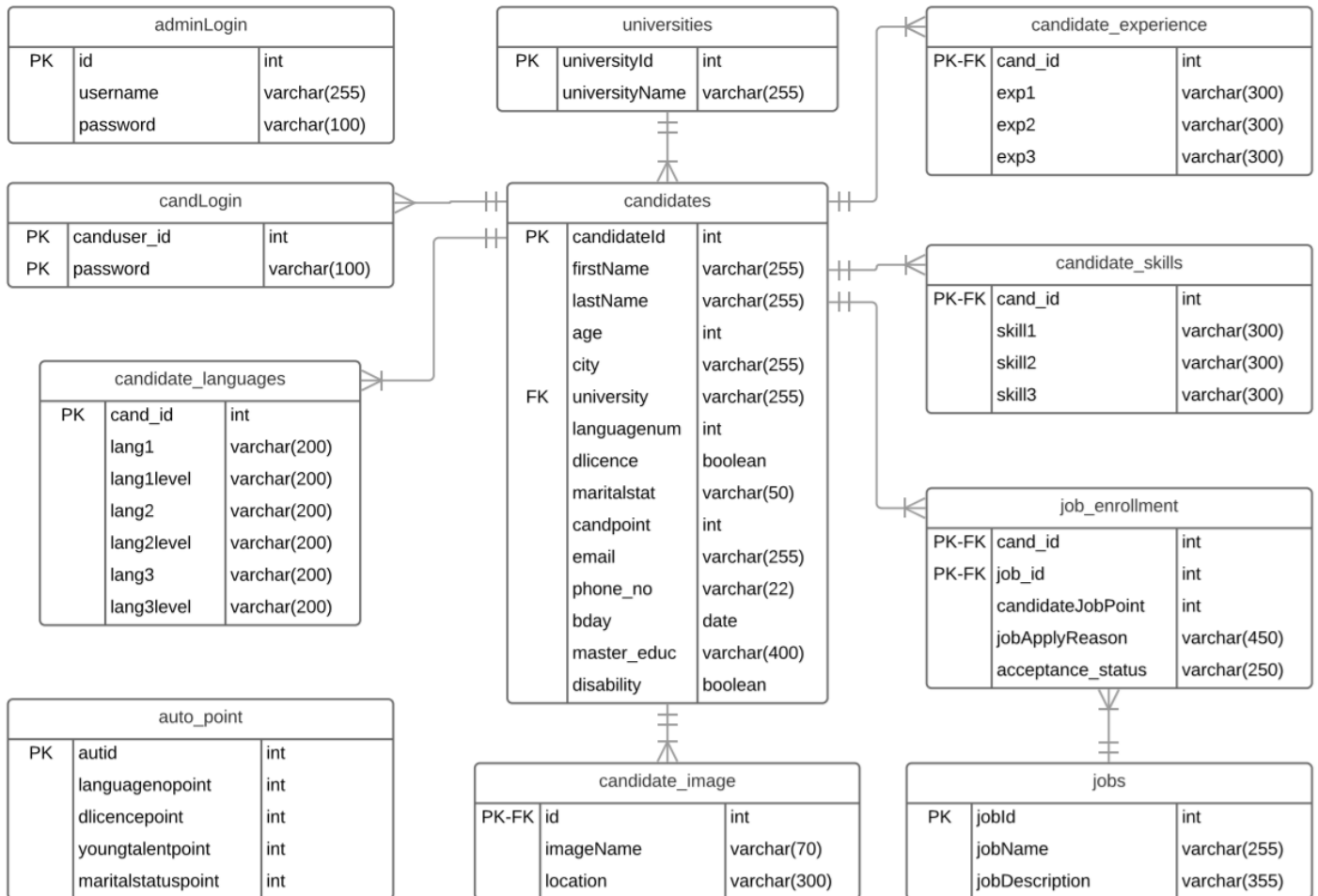


Figure [7]: Online Recruitment Application Database Model

As you can see in the model, **job\_enrollment** and **candLogin** entities have composite primary keys to avoid primary key violation. Relationship cardinality types are shown on the model. University candidate relationship is a has relation. Candidates, job enrolment and jobs relationships are apply relation. Candidates and candLogin entities relationship is also a has relation. Also in the database there are 4 views. One is **applied\_candidates** which shows candidates only who apply at least 1 job advertisement. The rest are created for skills, experience and language tables to make it easier to implement live search queries.

- **Object Character Recognition Implementation**

Tesseract is an Object Character Recognition Engine which success rate depends on quality of input image. Basically, it takes image file and recognize letters in this file so that it gives machine readable text as an output. Tesseract project does not have user interface it executes from command line interface. Tesseract share libtesseract with developers to build their own application. In this project there are several language options which can be use after including language trained data. When searching Turkish language trained data, I encounter with too many problems. Some trained data, don't give acceptable output, for example one of the trained data hasn't got letter "Ü". In Candidate Picker Project by typing "Üniversite" candidates use it a lot. After finding older version of Turkish Trained Data I solve this problem. Detailed explanation about Tesseract can be found on related work part of this report.

To set up Tesseract on Java, I use combined version Tess4J and Lept4J wrappers. I need these wrappers because libtesseract supports only C and C++ languages. After creating project on Eclipse, I take necessary jar files and dll files for win64 operating system. Also, I rearrange manifest file in project to prevent some errors. At the end I take Tesseract output and store it in txt file to use auto fill function.

- **Auto Fill Function**

Java part of the project is done by using Eclipse Photon. I started by making database connection with using referenced library which name is MySQL Connector Java 8.0.11. After arranging communication port number (3307) I setup connection between Java and MySQL.

On folderRead class there is a function which is filesInTheFolder. This function takes Tesseract output file and eliminates blank lines on the file. After this operation I stored lines into an Array. Next part I search keywords in this array elements. For example, when I find university keyword I start the search MBA, Master or Bachelor keywords which is for specifying Master's Degree or Bachelor Degree Education. Also, I search for Technical keyword for example Yıldız Technical University or İstanbul Technical University. I want to make sure while taking university name because my database has university table if there is a mistake MySQL gives constraint error. At the end I take only university name and store it related candidate's database tuple.

By taking my supervisor's advice I add experience and professional skill fields to candidate information area. Candidate and admin can add or change these fields by using web application and also using CV Auto Fill function. It has same working principle with University keyword search. While searching array we look for experience and related

keywords. After implementing pattern algorithm, we can successfully determine and take last 3 experience of a candidate. Also, skills and Expertise keywords search works like this. We take first 3 skills mentioned in CV and store it in candidate\_skills table.

Another candidate information area is Languages, by using Java we also take candidates languages and store it in relevant database table. I try to take language levels and store it in database column, but CV patterns can be changeable for example one type of CV mentioned level next to language information (English-Intermediate). Other type of CV mentioned level under the language name and write level like this “Full professional proficiency”. When I try to search these keywords program can take professional skills or another wrong information on CV.

Let’s look candidate contact information which is Telephone Number and Email Address. By using Java, we take this information and save candidate info table on our database. This transaction is done by searching “@” keyword for email; “+90”, “gsm” or “phone” keywords for telephone number. Figure [70]

When we consider CV types and compatibility issues, one of the biggest challenge shows up. Candidate’s CV don’t have certain pattern there are lots of different CV types. After my research, most used CV pattern is output of LinkedIn’s save PDF option. I decide the work with this pattern and 2 more different patterns. At the first I create select option for CV type, it increases success rate, but web page response time increases too. Another reason is there are too many types of CV’s I cannot create select option for all of them. Now software determines CV type by searching LinkedIn keyword in the CV. Because if candidate export CV from LinkedIn there will be LinkedIn Footer on the CV. At the end of all practises I create general algorithm which can work with most of the CV’s pattern.

- **Java-Web Site Integration**

It was real challenge for me, to encounter with this problem I search different methods to connect Java and Web Page. First, I try to create Java Applet and use it in my Web Application. But Java Applet Class is deprecated in JDK 9[12]. Because of the safety and performance issues. Another solution is to learn JavaFX and make connection by using it. But after I learn JavaFX [13] I cannot implement Tesseract Project on JavaFX. Also, I search for JSP and usage of Servlets. But I didn’t use this effectively. After than I discover Java-PHP Bridge which is what I want for this project. In the first experiments, it didn’t work. After changing Java.ini file and adding some log files I solve my problem and make this bridge work. Simply it works when I run my Java projects main class, Java-PHP Bridge open

connection between Java and Php with using 8080 port. Once I run my main class, it waits for Php requests.

- **Tools and Techniques**

While developing this system, I have used several different tools and techniques. Database creation and management is handled with SQL, using MySQL Workbench. To edit Web pages, I used Notepad++. Main page created with using HTML. Database connection and other important transactions handled with using PHP and AJAX. Styling HTML document with using CSS. While adding functions to HTML, I used JavaScript. To handle database transactions, I used PDO in order to manage and update the database by executing queries within the codes. Object Character Recognition and Auto-Fill Transactions is done with Java using Eclipse. For testing the project Selenium Browser Automation Software is used.

- **Technical, Operational and Financial Feasibility**

1. **Technical Feasibility**

I use software's newest versions in my implementation. In case these versions are updated, then I would have to modify my code and add new capabilities to my project.

2. **Operational Feasibility**

In companies Human Resources' organizational structure must be change, after usage of this software. Some of the employee can change or trained for gain specific computer skills to satisfy required knowledge.

- **Knowledge and Skill Set**

All the courses I took at Ozyegin University have contributed my development in several ways. The following courses are the ones that I have especially used our acquired knowledge in this project:

**CS102 – Object Oriented Programming:** This course enabled me to think and form algorithms in an object-oriented manner.

**CS202 – Database Management Systems:** Knowledge acquired from this course helped me to easily develop and manage a database.

**CS222 – Programming Studio:** On validation part of the project, this course helped me how to write tests which covers all corner cases.



**CS320 – Software Engineering:** This course helped me in the documentation and testing process of our project.

**CS333 – Analysis of Algorithms:** Just as in the case of Data Structures and Algorithms course, this course also provided me the knowledge and skill set to form efficient algorithms. It also helped me to think in a much more mathematical manner.

**CS447 – Computer Networks:** This course helped me to understand basics of Internet applications. Also, thanks to this course I consolidate how data communications between computer and server-side works.

- **Engineering Standards**

The following are the engineering standards I took into consideration while developing this project:

1. SQL:2016 (Structured Query Language) - ISO/IEC 9075:2016
2. PSR-1 (Basic Coding Standard)
3. PSR-2 (Coding Style Guide)
4. PSR-5(PHP-Doc)
5. HTTP (Hypertext Transfer Protocol)

## V. Results and Discussion

This project has two user types which are Admin and Candidate. In the application side of this project web pages and components are:

- User can login system if user has account. If user does not have any account, also can register and create new account. [Figure 9-11]
- When user enter wrong username or password. There is a warning message to warn the user about error. [Figure 10]
- Admin can select candidate and see selected candidate's information. [Figure 14-15]
- Admin can add new candidate to the system. User warned the result of this transaction by printing success message or error message. [Figure 16-19]
- When adding new candidate to the system if admin choose language number option as three. Form structure will be changed, three language and language level labels will be added. Experience and Skills labels works like this feature. [Figure 17-18]
- Admin can delete existing user by choosing from the list and get transaction result. [Figure 20-21]
- Admin can update candidate job acceptance status by choosing candidate and job which candidate apply before. After this admin can view current status and update status. Admin warn the result of this transaction. When status changed to Accepted, Candidate gets Information mail about interview. [Figure 22-23-24]
- Admin can handle live search based on Candidate Information. For example, admin wants to search candidates which location is in Eskisehir, by clicking city button and writing to wanted String, candidates will show up on key up. [Figure 25-26]
- Admin can print candidate Information by using show candidate option which is located in search candidate page. [Figure 27]
- Admin can add new job advertisements to the system. [Figure 27-28]
- Admin can list applied jobs by choosing related job. After this, admin view which candidates apply this job. Also, current candidate point can be changed on this section. When admin wants to change candidate point to ease decision making there is show pdf part which shows related candidate CV on new tab. [Figure 27-28-29-30]
- Admin can view automatic evaluation point and there is candidate ranking table which is descending point order. This ranking table is dynamic it updates every minute. [Figure 31]

- Admin can change automatic evaluation point setting by entering new point values to the system. [Figure 32]
- After login, Candidate can list current job advertisements. [Figure 35-36]
- Candidate can apply job advertisements and information mail will be sent to candidate's registered email address. [Figure 37-38-39]
- On the upload CV function, candidates can upload their CV's to the system. Transaction result share with candidate with using Success or Error message. If result is successful, there is another option for candidates which name is auto-fill. [Figure 40-41-42]
- Auto-Fill Function takes uploaded CV and send to the Separation Engine. As I mentioned before, first step is OCR to convert PDF or image, machine readable text. After this Java Parsing Algorithm takes important keywords arrange them and store into the candidate database's related sections. As you can see on the picture (Figure 43) output shows updated Candidate Information. In addition, execution time is for inform users.
- Candidate can update their personal information from Candidate Settings section. [Figure 43]
- Candidates can follow their job acceptance status. [Figure 44]
- Server setup is done. Ubuntu [14] and necessary software's installation completed. Necessary software are Apache, MySQL, Pdo Driver, PhpMyAdmin.

Validation of program is done by using 2 different methods. First one is testing which is done by using Selenium [15] and manually function testing done by hand.

Browser Compatibility Testing				
Test Name	IE	Chrome	Firefox	Edge
Page Structure	✓	✓	✓	✓
Button Styles	✓	✓	✗	✓
Export Table as PDF	✓	✓	✓	✓
Export Table as Excel	✗	✓	✓	✗
Menu Bar Functionalities	✓	✓	✓	✓

Table [3]: Compatibility Test Results

As you can see on the table [3], testing will be done by using different browsers. Because web site features can be change. Button styles, selection panes structure, padding options can be different. Also, some of the JavaScript packages does not supported by browsers. For example, Export Table as Excel feature did not work IE and Edge.

Functionality Testing Results		
Test Name	Success	Failure
Login Testing	✓	
Prevention of Login Bypass	✓	
Mobile Browser Usage		✗
Live Search Style and Output	✓	
Candidate Information Mail Feature	✓	
Candidate Ranking Feature Update	✓	
Candidate Profile Update	✓	
Upload CV and Auto Fill Function	✓	
Server Situation and Usability		✗

Table [4]: Functionality Test Results

Functionality testing part of the validation, first step is login testing. It done by using wrong characters and Strings can cause SQL Injection. Second test is about prevention of login bypass which means reach specific page with using page's URL. For example, candidate learn candidate ranking feature's URL. He can try to reach by entering this URL. But Session control feature on website controls user name and password are not null on all admin and candidate pages. Mobile browser usage testing is done by hand. Page Structure can be broke. It can be solved by using Bootstraps on the project. Candidate ranking feature update is a dynamic web page which refresh's table every minute. For testing this I add new candidate on this time interval. After the refresh this candidate shows up on the table. Also, I check dynamic web pages with using IE's developer tools part. Auto Fill Function testing is quite important for this project. As I mentioned before I use 3 different CV patterns. When these patterns are used to test this project, results are quite good. But always there can be wrong result of the OCR or parsing algorithm output it can be changeable for the different factors.

Selenium implementation is done by using IntelliJ Idea. First create maven project to get Selenium packages easily. After that I need to install web browser drivers to reach web site by using selenium test suit. On the below there is a test suite for Admin Login Case. First web driver object created. After that elements get by ID or Class name. In the picture we reach our elements by using their id's and we assign them to the correct login values which are Enes, 1995. At the end we get current URL after we click to the login button. Also I add some test cases' screenshots on the Appendix part of the report.

```

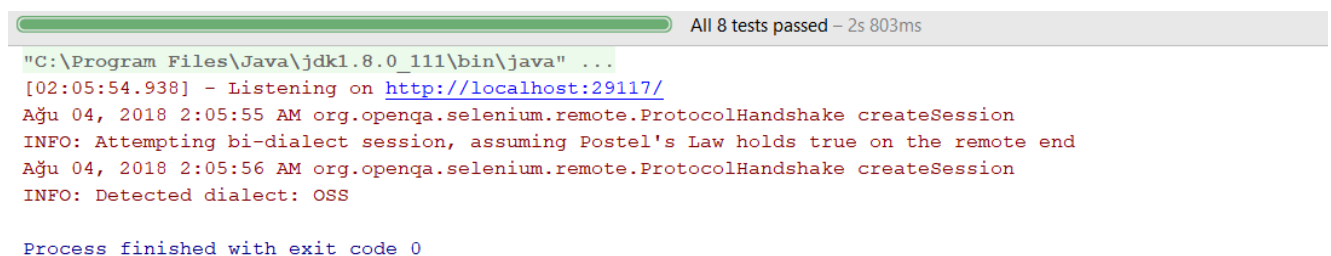
} public void CaseLogin() {
    System.setProperty("webdriver.edge.driver", "C:\\Users\\Hp\\Desktop\\selenium\\MicrosoftWebDriver.exe");
    WebDriver webDriver = new EdgeDriver();
    webDriver.get("http://localhost/adminlogin.php");

    WebElement nameLabel = webDriver.findElement(By.id("clicker"));
    nameLabel.click();
    nameLabel.sendKeys(...charSequences: "enes");
    WebElement passLabel = webDriver.findElement(By.id("pword"));
    passLabel.click();
    passLabel.sendKeys(...charSequences: "1995");
    WebElement button = webDriver.findElement(By.id("submit2"));
    button.click();

    String pageURL = webDriver.getCurrentUrl();
    assertEquals(pageURL, actual: "http://localhost/mainpage.php");
    //String pageTitle = webDriver.getTitle();
    // assertEquals(pageTitle, "Candidate Picker");
}

```

Figure [8]: Example Test Function Code



```

"C:\Program Files\Java\jdk1.8.0_111\bin\java" ...
[02:05:54.938] - Listening on http://localhost:29117/
Ağu 04, 2018 2:05:55 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Attempting bi-dialect session, assuming Postel's Law holds true on the remote end
Ağu 04, 2018 2:05:56 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS

Process finished with exit code 0

```

Figure [9]: Example Test Run

On the performance testing part of the validation, I check the response time of the features. At an average connection speed all page's response time under 1 second except auto fill feature. This feature works with Tesseract and Parsing algorithm. Average response time of this feature is 7,18 second. This average calculated by taking 10 different transactions response time. At the start Tesseract consumes nearly 3 seconds to give results. On the second step which works with parsing algorithm, it takes approximately 2 seconds to execute. At the end fill database tables using output of parsing algorithm takes 1.5 seconds. Also, Php-Java

Bride and Java-MySQL connection consumes little period of time. This result made by using 2-page standard CV format. It can be change variation of parameters such as PDF size, Image format, internet connection etc.

My supervisor suggested me to create survey and take user opinions about to program. I create a survey that covers program's general features. 20 user participate this survey, they use this program 10 minutes, after that they fill this survey. Also, I take their opinions about how this program can be more capable. Survey Questions and Average Points which is given by users shown on the table [5].

Questions		Average Points
1	Interface Usability	3
2	Response Time of the System	2,7
3	Idea of the Project	2,95
4	Interface Understandability	2,9
5	Candidate Ranking Feature	2,75
6	Auto-Fill Function	2,6
7	Admin Capabilities	2,8
8	Web Site Design	2,9
9	Candidate Capabilities	2,55
10	General Overview	3

Table [5]: Evaluation Survey Results

There are 3 main categories to answer all questions.

1-Bad

2-Improvable

3-Good

Summary of the survey results are quite good for my expectations. To general idea is project is good when some of the features are improved it will be better.

## VI. Related Work

### A) Related Projects

I did a research on existing Online Recruitment Applications before starting to the project. There are similar projects on the web. We can divide these projects into two categories. First one is intra-company recruitment applications which are developed only for their own company. For example, Turkcell Recruitment System or Google Careers. Second one is for multiple companies and this has too many areas to apply. For example, Kariyer.Net Hiring Application. Also, there is social networks for business which you can manage your professional identity and apply for a job. Example of this is LinkedIn.

When we look at the Google Careers [16], it has comprehensive search system based on Location, Job Type and Position. Also, Web Site has more filtering options which depends on Education, Experience and Type. While I am searching similar projects Google Careers is the most inspiring project for my research. This is the first part of the Google Career path which name is Apply. Second part is Interview and third part is Decide. In this project I will be interested in only Apply part and try to implement these capabilities to my project.

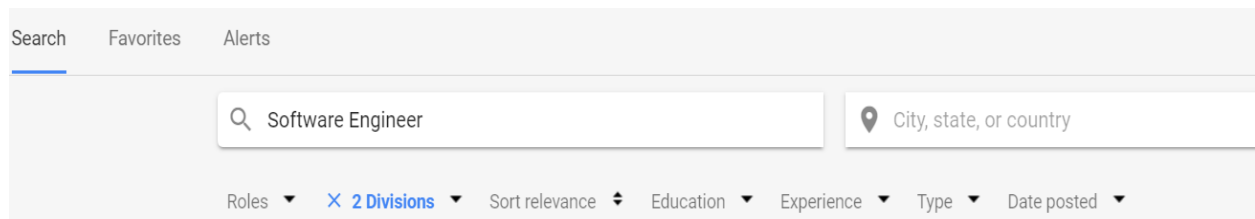


Figure [10]: Google Careers Search Example

If we look local Online Recruitment Website I think most productive one is “Turkcell Kariyer” [17]. This application there are available positions link which can you apply for a new job. Also, there has a search label with different filters. One the filter is search by using keywords. I will add this feature to my project. When I login to system as a candidate, there is my profile section. By using this section candidates can upload their CV’s to the system. Also, they create different parts of this section which candidate can enter own education reference and experience. I think they store this information to their database by using this part of program. My program will be take this information by investigating candidate CV’s by using separation algorithms. To do this CV’s must be similar pattern.

TURKCELL.COM.TR

**AÇIK POZİSYONLAR**

**Arama Yap**

**Kayıtlı Aramalarım**  

Seçiniz
▼

**Anahtar Kelimeler**

**Lokasyon**  

Seçiniz
Q

**Çalışma Alanı**  

Seçiniz
Q

**İş Tipi**  

Seçiniz
Q

Kaydet

Arama

**ARAMA SONUÇLARI**

**11128 - BÖLGE YÖNETİCİLİĞİ**  

•Radyo şebeke planlama tarafından gele  
etiği içinde yönetmek ve gerekli görüşm

[Detaylar >](#)

**11149 - SENIOR DATA ANALYTICS EN**  

İşlettiği servis ve platformların günlük/ç  
operasyonel kalite standartlarına uygun

[Detaylar >](#)

**11228 - SENIOR HARDWARE&TECHN**  

Tedarikçi seçim sürecinde ticari seçim kr  
sözleşmelerin yönetilmesi Anten, iklimle

[Detaylar >](#)

**11084 - SALES ACCOUNT MANAGER**  

Kobiler ile olan ilişkilerin yönetilmesi ve

Figure [11]: Turkcell's Recruitment System Example

If we look at the second category we can easily say that, Kariyer.Net [2] well-known local Recruitment Website. When we criticize, site has very complicated main page and also it forces candidates to fill all information before you apply any job. I think this causing the candidate to get bored and candidate may give up to fill all information by this reason cannot apply the job he want. In my project when applying a new job, I will decrease candidate effort to the minimum level. If we look Kariyer.Net's pros, after the fill all information candidates can apply different companies. It is good feature for candidates, however it complicate employer's hiring process. Because employer needs to check different channels all time.

İş Ara 
Kariyer Akademi
Blog

ÜYE OL
ÜYE GİRİŞİ
İŞVEREN ▼
İLAN YAYINLA

Senin için burada 104.242 ilan var!

Şehir
▼

Pozisyon, firma adı, sektör

İŞ BUL

[DETAYLI ARA](#)

**ÖNE ÇIKAN İŞ İLANLARI**

**ÜRETİM ELEMANI**  
 Toyota Otomotiv San...  
**TOYOTA**

**AKASYA BOYNER ...**  
 Boyner Holding (1102)  
**BOYNER**

**Tahsilat Temsilcisi**  
 Güven Varlık Yönetim ...

**Call Center Agents...**  
 Bir Telecom  
**BİR TELECOM A.Ş.**

**UI Developer**  
 Siemens San. ve Tic. ...  
**SIEMENS**

**Turkish Speaking ...**  
 Apple

Figure [12]: Kariyer.Net Main Page Example



## **B) Object Character Recognition Working Principle**

When we take photo or scan a document, output is picture format on our computer. Computer can not recognize letters on the picture. If we want to edit or search some keywords on the output, it may be difficult because we need to examine all pages. To solve this difficulty OCR technology is used. Simply OCR reads image line by line and try to detect letters. This detection can be done using various algorithms on different recognition stages. For example, some of the experts suggests Deep Neural Network algorithm for handwritten inputs. We can study this technology by dividing three main parts. They are Pre-processing, Character Recognition and Post Processing. On the pre-processing part program prepares image to operate. This preparation can be made by adjusting size and contrast, clear unnecessary signs and letters will be sharpened to increase readability. After this second part program scans shapes and try to match these shapes with Training Data's Alphabet and defined fonts. On the last step OCR try to guess words using Training Data. After this program outputs the document as machine readable text. [18]

There are several OCR software's in the market. I search free, open source and robust packages to use on my Web Site. In my options some of them did not support Turkish Language and also some of them needs to be operate with using Python. I choose Tesseract because when I examine projects which developed using Tesseract, there are convincing comments on the reports. Also, if we compare Tesseract's capabilities to another free OCR software, we can easily see Tesseract is the winner for this project. It is developed by Google and HP. Tesseract supports 102 languages and 4500 fonts on Version 4.0. [19]

## VII. Conclusion and Future Work

In conclusion, I have a capable database that keeps candidate, admin and job advertisement information. Finally, Web site has features like login, registration, admin and candidate panels with various capabilities.

- **Impacts of The Project**

Companies can increase efficiency of their employee with using this software. In company employee gaps will be filled quickly. When we look at the social side, recruitment process will be shortened than old school hiring process. Thanks to this, candidates will be more motivated because they can learn acceptance situation faster. If there is rejection situation candidate time loss will be less. Economic impact of this project is companies set to work less employee on Human Resources team. By this way company expenses will decrease.

Another important impact is related with Turkish Business Laws. From 2018, companies which have more than 50 employees must hire disabled personal. By using Online Recruitment Application, companies can easily choose disabled person using auto point function. [20]

When we consider ethics of this project, in this project I need personal information of candidates which is their CV information. Another ethical effect is all candidates' CV's will be equal there will be no friend at court situation in the company.

- **Effects of The Project**

- a) **Sustainability:**

- This project is sustainable since it can be developed further by improving website and applying different algorithms.

- b) **Manufacturability:**

- The system is easily manufacturable because of its software nature. Project can be adapted to various sectors.

- c) **Health:**

- There is not any health impact or constraint of this project.

- d) **Security:**

- The user agreement will cover the security of personal information. There is no physical security risk or constraint.

This senior project I try to increase efficiency on hiring process. I take my colleagues and supervisor's ideas to enhance application's features. I learned too many things in this period. I can list some of the learning outcomes of this project:

1. Have used OCR technology and learned basics of image processing.
2. I practised many different programming languages.
3. I learned how to integrate PHP-Java and MySQL.
4. Created a server, learned communication through ports and use droplet functions effectively.
5. Worked with using several protocols (SMTP-IPV6-SSL-FTP)
6. Learned how to use JavaScript, AJAX and create dynamic web pages.
7. Worked with different Operating Systems (CentOS, Ubuntu)
8. Learned hiring process and company's responsibilities while this process.
9. I learned how to prevent security breaches while creating Web Page.
10. Learned how to use Selenium and which techniques are used while testing the web application.

On the second part of the project (CS 402) listed features is implemented and studied:

1. Proper OCR and OCR working principle research.
2. Parsing Algorithm is created by observing general CV types.
3. Java-PHP Bridge Setup
4. Search University Information is changed to Search Candidate Information. Now these search feature works for all candidate information.
5. New Candidate Information areas (Skills-Experience etc.) added to the Program.
6. Candidate Ranking feature structure changed to dynamic.
7. Candidate profile settings, candidate status, send mail functions has new features.
8. Table Export as PDF or Excel Feature is added.
9. Exception's structure changed now transaction messages are more understandable.
10. Front-end Capabilities are enhanced.
11. Bug detection and software improvement done.
12. Software validation is done by taking user ideas and some of the requirements tested by using Selenium and manually.

- **Future Work**

There are some possible improvements which can be done in future. The following can be for future work:

1. Bootstrap can be used while styling the web site. This will increase compatibility of the project. By this way mobile browser structure will be more understandable.
2. Foreign universities can be added to the University options.
3. Skills and Experience areas can be categorized. For example, computer skills, communication skills.
4. More efficient OCR can be implemented.
5. CV parsing algorithm can be enhanced.
6. Mail option can be change by adding mail content option to the admin panel.

## **Acknowledgements**

I would like to thank Erhan ÖZTOP and Tankut Barış AKTEMUR for their help and support on project.

I would also like to thank all our professors in Computer Sciences Department.

I would thank to all survey participants to their help on the validation of this project.

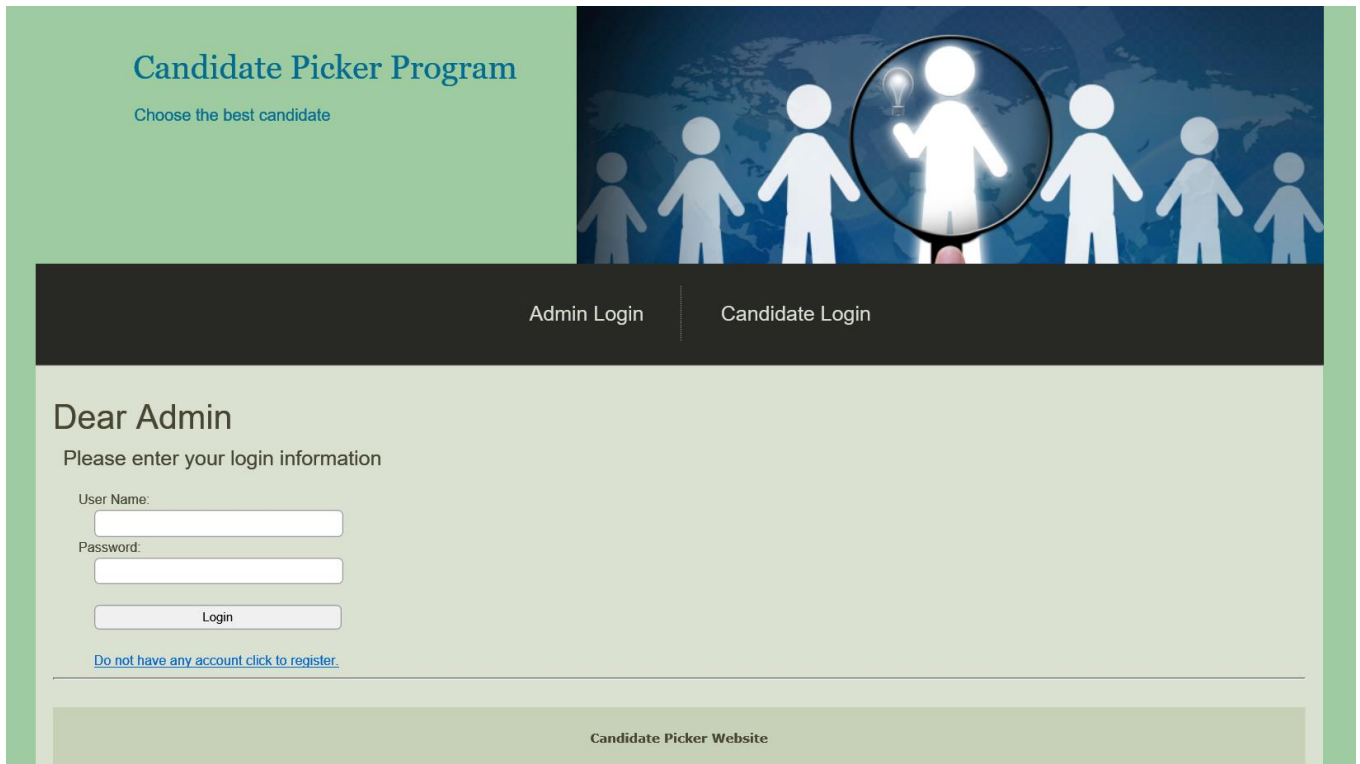
I would like to thank for my old colleagues who inspired me to create this project.

## References

- [1] <https://www.prevuehr.com/resources/insights/how-long-should-a-hiring-process-take/>
- [2] <https://www.kariyer.net/>
- [3] <https://www.linkedin.com/>
- [4] <https://www.digitalocean.com/>
- [5] <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>
- [6] <https://github.com/PHPMailer/PHPMailer/tree/5.2-stable>
- [7] <http://php-java-bridge.sourceforge.net/doc/desktop-apps.php>
- [8] <https://opensource.google.com/projects/tesseract>
- [9] <https://github.com/MrRio/jsPDF>
- [10] <https://code.jquery.com/jquery/>
- [11] <http://php.net/manual/tr/book.pdo.php>
- [12] <https://www.oracle.com/technetwork/java/javase/9-deprecated-features-3745636.html>
- [13] <https://docs.oracle.com/javafx/2/overview/jfxpub-overview.htm>
- [14] <https://www.digitalocean.com/community/tutorials/initial-server-setup-with-ubuntu-16-04>
- [15] <https://www.seleniumhq.org/>
- [16] <https://careers.google.com/>
- [17] <https://kariyer.turkcell.com.tr/S/JobSearch>
- [18] <https://anyline.com/news/what-is-ocr/>
- [19] <https://github.com/tesseract-ocr/tesseract/wiki/TrainingTesseract-4.00>
- [20] <https://www.tbmm.gov.tr/kanunlar/k4857.html>

## Appendix

In this chapter, detailed description such as codes, website view, and screenshots can be found.



The screenshot displays the 'Candidate Picker Program' interface. The header features the program name and the tagline 'Choose the best candidate' on a green background. To the right is a graphic of stylized human figures, with one figure highlighted by a magnifying glass and a lightbulb icon. Below the header is a dark navigation bar with 'Admin Login' and 'Candidate Login' links. The main content area, titled 'Dear Admin', prompts the user to enter login information. It includes input fields for 'User Name' and 'Password', a 'Login' button, and a link to register for new users. The footer identifies the page as the 'Candidate Picker Website'.

Candidate Picker Program

Choose the best candidate

Admin Login | Candidate Login

Dear Admin

Please enter your login information

User Name:

Password:

[Do not have any account click to register.](#)

Candidate Picker Website

Figure [13]: Admin Login Page



The screenshot shows the 'Transaction Result' page following an unsuccessful login attempt. It features a light green background with a message indicating that the 'Invalid username and password!' error has occurred. The footer of the page is labeled 'Candidate Picker Website'.

Transaction Result

Invalid username and password!

Candidate Picker Website

Figure [14]: Admin Unsuccessful Login Result

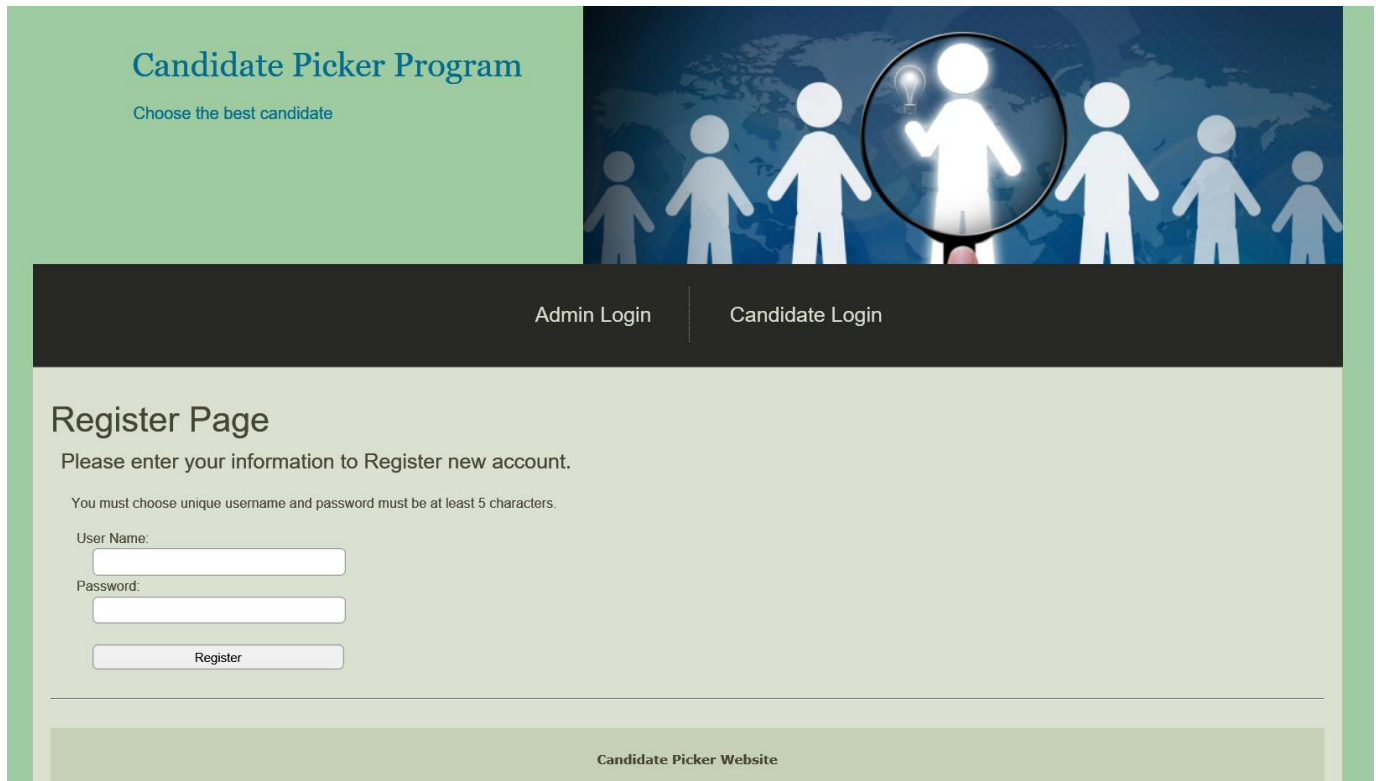


Figure [15]: Admin Register Page

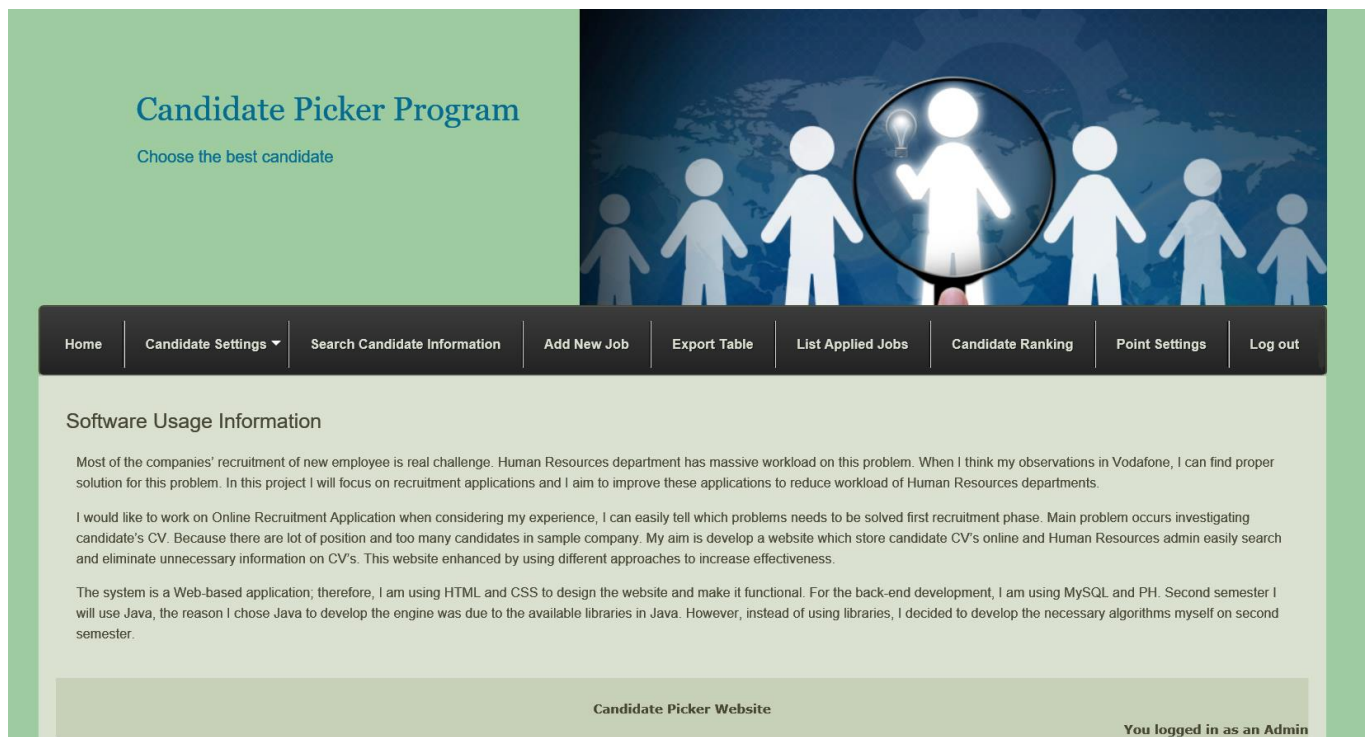


Figure [16]: Admin Panel Main Page



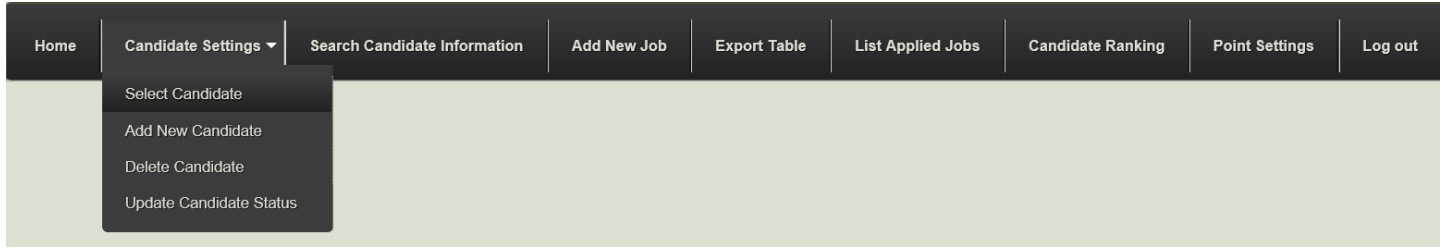


Figure [17]: Admin Panel Menu Bar

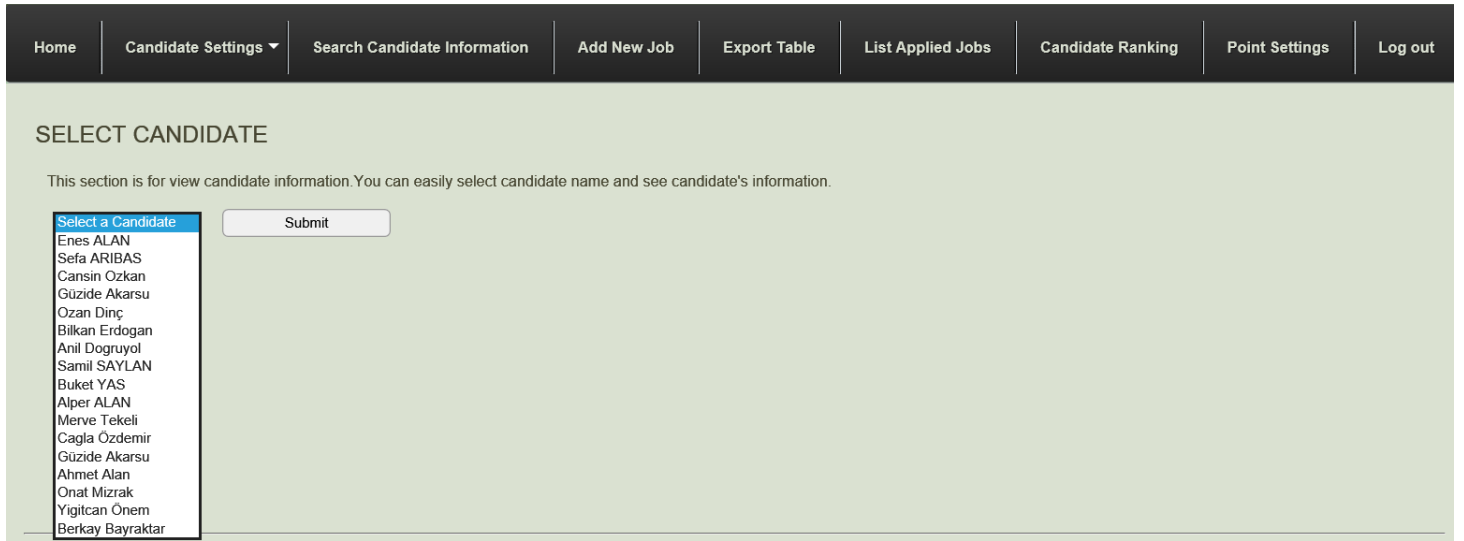


Figure [18]: Admin Panel Select Candidate Page



Figure [19]: Admin Panel Selection Result Page

## ADD NEW CANDIDATE

In this section, you can add new candidate. Please fill all labels.

Candidate Id:

First name:

Last name:

City:

E-mail:

Phone Number:

Birthday:

Foreign Language Number:

1.Language:

Driving Licence:

Marital Status:

Disability:

Please Choose number of your Experience which you want to share:

1.Experience:

Please Choose number of your Professional Skills which you want to share:

1.Skill:

University:

Master Education:

Figure [20]: Admin Panel Add New Candidate Page

Foreign Language Number:

1.Language:

2.Language:

3.Language:

Figure [21]: Admin Panel Add New Candidate Language Part

Please Choose number of your Experience which you want to share:

2

1.Experience:

2.Experience:

Please Choose number of your Professional Skills which you want to share:

3

1.Skill:

2.Skill:

3.Skill:

Figure [22]: Admin Panel Add New Candidate Experience and Skills Part

## Transaction Result

Candidate Kuzey ALAN Succesfully Added

Candidate Picker Website

Figure [23]: Admin Panel Add New Candidate Result Page

## DELETE CANDIDATE

This section is for delete candidate information.You can easily select candidate name and delete candidate.

Select a Candidate

Enes ALAN

Sefa Aribas

Cansin Ozkan

Guzide Akarsu

Alper ALAN

Hatice Ertural

Ozan Dinc

Bilkan Erdogan

Anil Dogruyol

Submit

Candidate Picker Website

You logged in as an Admin

Figure [24]: Admin Panel Delete Candidate Page

## DELETED CANDIDATE

Deleted Candidate information is shown below.

ID	Firstname	Lastname	Age	City	Candidate Education Information
119	Merve	Tekeli	21	Eskişehir	ANADOLU ÜNİVERSİTESİ

Candidate Picker Website

You logged in as an Admin

Figure [25]: Admin Panel Delete Candidate Result Page

## Update Candidate Acceptance Status

In this section you can change candidate acceptance situation.  
First you need to choose which candidate status wants to be changed.

Select a Candidate  
Enes ALAN  
Cansin Ozkan  
Güzide Akarsu  
Sefa ARIBAS  
Bilkan Erdogan  
Samil SAYLAN

Submit

Candidate Picker Website
You logged in as an Admin

Figure [26]: Admin Panel Update Candidate Status Candidate Selection Page

## UPDATE CANDIDATE STATUS

Selected Candidate information is shown below.

ID	Firstname	Lastname	Age	City	Candidate Education Information
102	Cansin	Ozkan	25	Bursa	YILDIZ TEKNİK ÜNİVERSİTESİ

This candidate applied these jobs

Job ID	Job Name	Status	Select Status	Change Status
1	CRM Solutions Senior Specialist	Rejected	Select a Status	Change
2	Billing Services Operations Senior Specialist	Accepted	Select a Status	Change
3	IP CPN 1st Level Operations Specialist	Analyzing	Select a Status	Change

Candidate Picker Website
You logged in as an Admin

Figure [27]: Admin Panel Update Candidate Status Page

## Transaction Result

Interview Content Mail sent to your registered mail address

Successfully changed current candidate status " Analyzing " to " Accepted " for candidate id 100.

Candidate Picker Website
You logged in as an Admin

Figure [28]: Admin Panel Update Candidate Status Result Page

SEARCH CANDIDATE

This section is for view candidate information. You can easily select candidate name and see candidate's information.

University

City

Age

Language

Skills

Master Education

Experience

Please type part of the String:

eski

×

ÖZYEĞİN ÜNİVERSİTESİ

Candidate ID:100   Candidate Name : Enes ALAN

Show Candidate

SABANCI ÜNİVERSİTESİ

Candidate ID:104   Candidate Name : Ozan Dinç

Show Candidate

AKDENİZ ÜNİVERSİTESİ

Candidate ID:108   Candidate Name : Anil Dogruyol

Show Candidate

ÖZYEĞİN ÜNİVERSİTESİ

Candidate ID:110   Candidate Name : Samil SAYLAN

Show Candidate

YILDIZ TEKNİK ÜNİVERSİTESİ

Candidate ID:115   Candidate Name : Buket YAS

Show Candidate

ANADOLU ÜNİVERSİTESİ

Candidate ID:117   Candidate Name : Alper ALAN

Show Candidate

AKDENİZ ÜNİVERSİTESİ

Candidate ID:127   Candidate Name : Kuzey ALAN

Show Candidate

Candidate Picker Website

You logged in as an Admin

Figure [29]: Admin Panel Search Candidate Page

SEARCH CANDIDATE

This section is for view candidate information. You can easily select candidate name and see candidate's information.

University

City

Age

Language

Skills

Master Education

Experience

Please type part of the String:

german

×

SABANCI ÜNİVERSİTESİ

Candidate ID:139   Candidate Name : Onat Mizrak

Show Candidate

EGE ÜNİVERSİTESİ

Candidate ID:141   Candidate Name : Berkay Bayraktar

Show Candidate

Candidate Picker Website

You logged in as an Admin

Figure [30]: Admin Panel Search Candidate Page-2

## Candidate Information Print Page

ID:100

Firstname : Enes

Lastname : ALAN

Age : 23

Phone Number : 5426299228

### Education

Bachelor Degree : ÖZYEĞİN ÜNİVERSİTESİ

Master Degree : Harvard University

### Personal Information

City :Eskisehir

Marital Status:single

Driving Licence : Yes

Birthday : 1995-10-29

### Languages

English

French



Print This Page

Figure [31]: Admin Panel Show Candidate Print Page

### ADD NEW JOB

This section is for add new job to the system.You must enter job id,job name and job description to create new job advertisement.

Job Id:

Job name:

Job Description:

Submit

Candidate Picker Website

You logged in as an Admin

Figure [32]: Admin Panel Add Job Advertisement Page

## Transaction Result

Job Computer Engineer Successfully Added

Candidate Picker Website

Figure [33]: Admin Panel Add Job Advertisement Result Page

### CHOOSE A JOB

This section is to view candidates who apply selected job.

Please Choose a Job:

Select a Job

Submit

Candidate Picker Website

You logged in as an Admin

Figure [34]: Admin Panel List Applied Jobs Page

### CHOOSE A JOB

This section is to view candidates who apply selected job.

Please Choose a Job:

Select a Job

CRM Solutions Senior Specialist  
Billing Services Operations Senior Specialist  
IP CPN 1st Level Operations Specialist  
test  
Data Engineer  
Manager  
Computer Engineer

Candidate Picker Website

You logged in as an Admin

Figure [35]: Admin Panel List Applied Jobs Page-2

### Now Listing Candidates for Job:

#### CRM Solutions Senior Specialist

This section is to view candidates who apply selected job.

By using this section admin can also change candidate points to clicking submit button.

ID	Description	Current Candidate Point	Enter New Candidate Point	Change Candidate Point	Look Candidate CV
100	I didn't really connect the dots between the major and potential career until my Senior year.I realize this job is for me.	5	<input type="text"/>	<button>Change</button>	<a href="#">Show Pdf</a>
102	I want this job.	4	<input type="text"/>	<button>Change</button>	<a href="#">Show Pdf</a>
103	I want to do this job.	4	<input type="text"/>	<button>Change</button>	<a href="#">Show Pdf</a>

Candidate Picker Website

You logged in as an Admin

Figure [36]: Admin Panel Listed Applied Jobs Page

## Transaction Result

Successfully changed current evaluation point " 5 " to " 3 " for candidate id 100.

Candidate Picker Website

Figure [37]: Admin Panel Enter Candidate Point Result Page



### Candidate Automatic Evaluation Point Ranking

ID	Firstname	Lastname	Age	City	University	Candidate Point
141	Berkay	Bayraktar	19	İzmir	EGE ÜNİVERSİTESİ	63
127	Kuzey	ALAN	21	Eskisehir	AKDENİZ ÜNİVERSİTESİ	56
120	Cagla	Özdemir	20	Adana	MARMARA ÜNİVERSİTESİ	40
100	Enes	ALAN	23	Eskisehir	ÖZYEĞİN ÜNİVERSİTESİ	33
104	Ozan	Dinç	19	Eskisehir	SABANCI ÜNİVERSİTESİ	33
140	Yigitcan	Önem	21	İzmit	ANKARA YILDIRIM BEYAZIT ÜNİVERSİTESİ	33
139	Onat	Mizrak	22	Kirsehir	SABANCI ÜNİVERSİTESİ	32
115	Buket	YAS	21	Eskisehir	YILDIZ TEKNİK ÜNİVERSİTESİ	31
101	Sefa	ARIBAS	24	Aksaray	MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ	28
102	Cansin	Ozkan	25	Bursa	YILDIZ TEKNİK ÜNİVERSİTESİ	26
107	Bilkan	Erdogan	48	Adana	MARMARA ÜNİVERSİTESİ	18
103	Güzide	Akarsu	48	Kocaeli	MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ	15
110	Samil	SAYLAN	22	Eskisehir	ÖZYEĞİN ÜNİVERSİTESİ	15
117	Alper	ALAN	24	Eskisehir	ANADOLU ÜNİVERSİTESİ	12
108	Anil	Dogruyol	24	Eskisehir	AKDENİZ ÜNİVERSİTESİ	10

Candidate Picker Website

You logged in as an Admin

Figure [38]: Admin Panel Candidate Ranking Page

### Automatic Point System Settings Page

In this section, you can change automatic point system settings. Please fill all labels.  
Current Points is displayed next to the input labels.

Foreign Language:  Current Value : 5

Driving Licence:  Current Value : 2

Young Talent:  Current Value : 10

Marital Status:  Current Value : 1

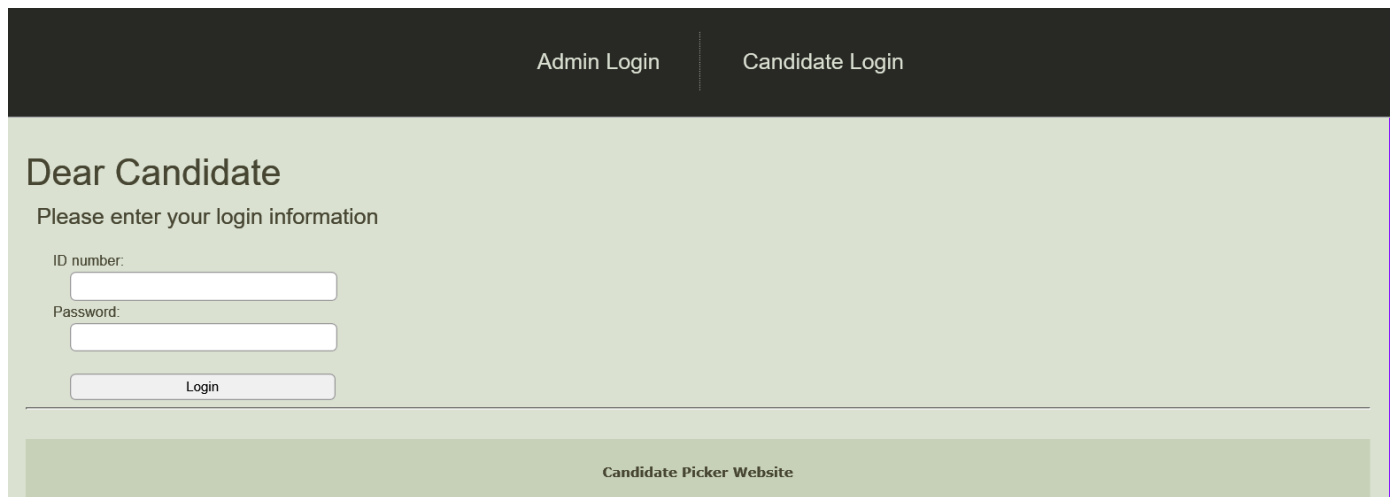
Master Education:  Current Value : 15

Disability Flag:  Current Value : 30

Candidate Picker Website

You logged in as an Admin

Figure [39]: Admin Panel Point Settings Page



The image shows a web page for candidate login. At the top, there is a dark header with two links: "Admin Login" and "Candidate Login". Below the header, the main content area has a light green background. It starts with the heading "Dear Candidate" followed by the instruction "Please enter your login information". There are two input fields: "ID number:" and "Password:". Below these fields is a "Login" button. At the bottom of the page, there is a footer with the text "Candidate Picker Website".

Admin Login | Candidate Login

## Dear Candidate

Please enter your login information

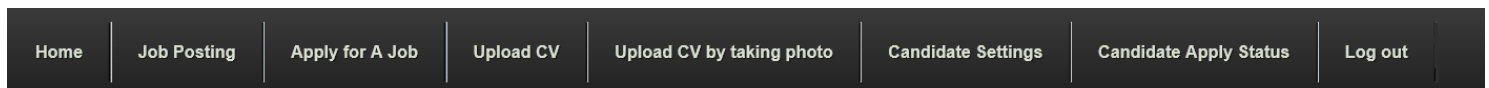
ID number:

Password:

Login

Candidate Picker Website

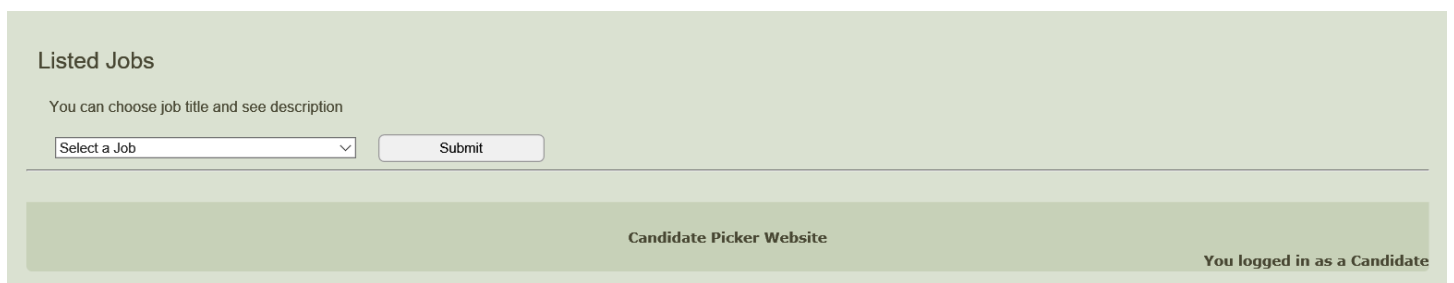
Figure [40]: Candidate Login Page



The image shows a horizontal menu bar with a dark background and white text. It contains eight items: "Home", "Job Posting", "Apply for A Job", "Upload CV", "Upload CV by taking photo", "Candidate Settings", "Candidate Apply Status", and "Log out". Each item is separated by a vertical line.

Home | Job Posting | Apply for A Job | Upload CV | Upload CV by taking photo | Candidate Settings | Candidate Apply Status | Log out

Figure [41]: Candidate Panel Menu Bar



The image shows a section of the candidate panel for job posting. It has a light green background. The section is titled "Listed Jobs" and includes the instruction "You can choose job title and see description". There is a dropdown menu labeled "Select a Job" and a "Submit" button. At the bottom of the page, there is a footer with the text "Candidate Picker Website" and "You logged in as a Candidate".

### Listed Jobs

You can choose job title and see description

Select a Job

Candidate Picker Website

You logged in as a Candidate

Figure [42]: Candidate Panel Job Posting Section

Listed Job

ID	Job name
3	IP CPN 1st Level Operations Specialist

Job Description

Make troubleshooting in case of faults or service outages. Find root cause of faults and take corrective actions Work in a proactive way to prevent incidents Give support to other teams.

Candidate Picker Website

You logged in as a Candidate

Figure [43]: Candidate Panel Job Posting Result

Apply Part

In this section, you can choose offered jobs and give small explanation why you choose this.

Please Choose a Job:

Data Engineer

▼

Why you choose this job:

Submit

Figure [44]: Candidate Panel Apply For a Job Section

Home

Job Posting

Apply for A Job

Upload CV

Log out

Congratulations

You succesfully apply job which Name is Computer Engineer

Candidate Picker Website

Figure [45]: Candidate Panel Apply For a Job Result



Figure [46]: Candidate Panel Apply For a Job Error Message

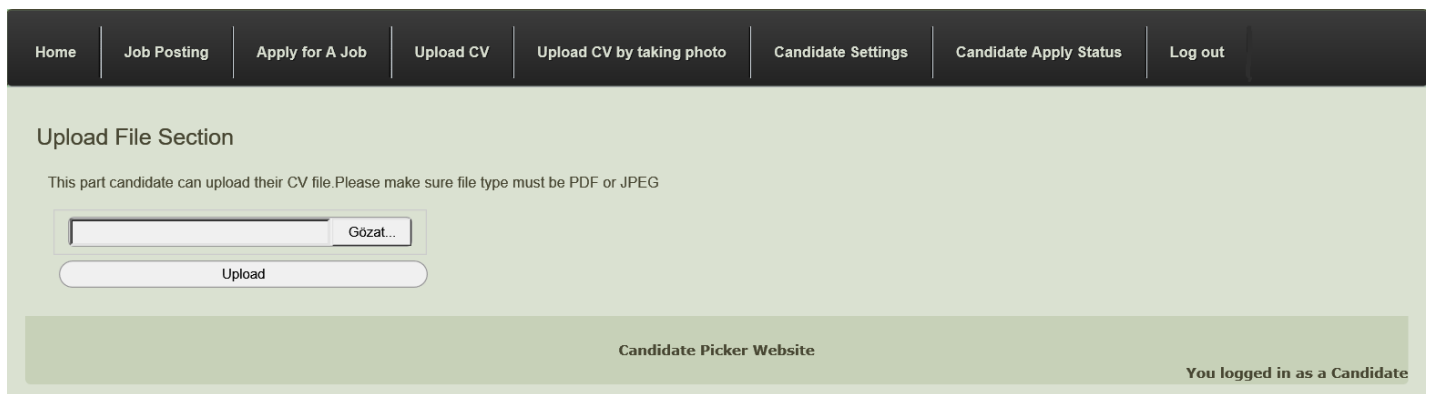


Figure [47]: Candidate Panel Upload CV Section

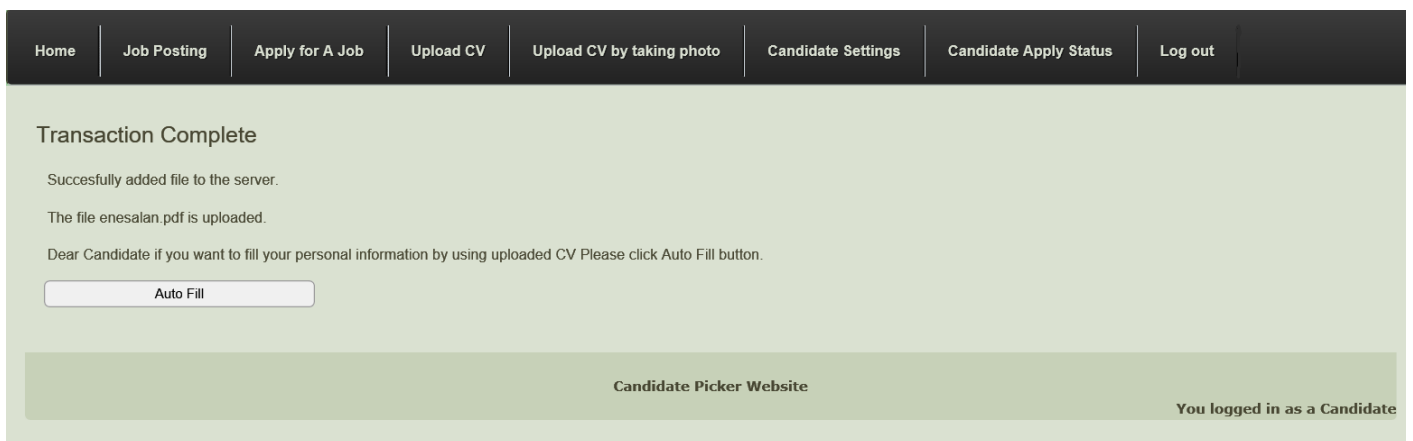


Figure [48]: Candidate Panel Upload CV Successful Result



Figure [49]: Candidate Panel Upload CV Error Message



Figure [50]: Candidate Panel Auto-Fill Function Output

## Cansın Özkan

HR Analytics Analyst - Vodafone

[cansinozk@gmail.com](mailto:cansinozk@gmail.com)

TEL : +905426299228

### Summary

N/A

### Experience

#### HR Analytics Analyst at Vodafone

Temmuz 2015 - Present

#### HR Intern at Vodafone Turkey

Kasım 2014 - Temmuz 2015 (9 ay)

#### Human Resources Intern at Koçtaş

Ağustos 2014 - Eylül 2014 (2 ay)

- Identified and approached suitable candidates,
- Learned to use databases to match the right person to the client's vacancy,
- Managed interviews and tests,
- Involved in informing candidates process about results.

#### Purchasing Intern at Teknosa

Haziran 2014 - Temmuz 2014 (2 ay)

- Worked in supply chain selection process,
- Observed price negotiations and price decisions,
- Had to chance to work with Oracle,

Figure [51]: Candidate Panel Auto-Fill Function Input

### Education

#### Yıldız Teknik Üniversitesi

3,13, Industrial Engineering, 2011 – 2015

#### Anadolu Üniversitesi

MBA-Business Relations

### Skills

Human Traffic

### Language

English-Beginner

French-Intermediate

Figure [52]: Candidate Panel Auto-Fill Function Input 2  
(Master University-Skills-Language areas added by me to test algorithm's functionalities.)

## Candidate Information Confirmation

Dear Candidate you can see registered information of yours, after the auto fill transaction.If there is a incorrect information you can go Candidate Settings Page and Update.

Execution Time : 7.2056088447571

First name : **Enes**

Last name : **ALAN**

City : **Eskisehir**

E-mail : **enes.alan@ozu.edu.tr**

Phone Number : **+905426299228**

Birthday : **1995-10-29**

Foreign Languages:

**English**

**French**

**Spanish**

Experience:

**VODAFONE- Istanbul, TURKEY (07/2017 - 11/2017)**

**EROGLU SOFTWARE - Eskisehir, TURKEY (07/2016 - 09/2016)**

**TTL TECHNOLOGY- Eskisehir, TURKEY (07/2015 - 08/2015)**

Skills:

o **MS Office, Unity, Intellij Idea, Wireshark,**

**cisco packet tracer, tableau, spss**

**Database: Oracle PL/SQL, MySQL**

Driving Licence: Yes

Disability: No

Master Education: **Harvard University**

Marital Status: single

University: **ÖZYEGİN ÜNİVERSİTESİ**

Figure [53]: Candidate Panel Auto-Fill Function Output Example 2

	<b>Ibrahim Enes ALAN</b>
	Address: 125. St. Num:87 Maslak, Istanbul/TURKEY
	E-mail: <a href="mailto:enes.alan@ozu.edu.tr">enes.alan@ozu.edu.tr</a> Mobile: +905426299228
<b>OBJECTIVE</b>	♦ To obtain an internship position specialized in technology at a reputable company
<b>EDUCATION</b>	<b>ÖZYEGİN UNIVERSITY</b> - Istanbul, TURKEY (2013 - Present) Faculty of Engineering Computer Science Department Year: 5 <sup>th</sup> Honors: 50% Exam Scholarship Cumulative GPA: 2.37/4.00  <b>KILICOGLU ANATOLIAN HIGH SCHOOL</b> - Eskisehir, TURKEY (2009 - 2013)
<b>EXPERIENCE</b>	<b>VODAFONE</b> – Istanbul, TURKEY (07/2017 - 11/2017) Position: Intern, HR Department ♦ HR Analytics Team ♦ Creation of Vodafone Employee Database ♦ Analyze Employee CV Information  <b>EROGLU SOFTWARE</b> - Eskisehir, TURKEY (07/2016 - 09/2016) Position: Intern, R&D Department ♦ Eroglu Companies User Interface and Database Creation  <b>TTL TECHNOLOGY</b> - Eskisehir, TURKEY (07/2015 - 08/2015) Position: Intern, R&D Department ♦ Hospital Database System Creation ♦ TURMEPA Museum Presentation Game Project ♦ Intel Atom Computers Setup

Figure [54]: Candidate Panel Auto-Fill Function Input Example 2

SKILLS & CERTIFICATES	TECHNICAL SKILLS	CERTIFICATES
	<ul style="list-style-type: none"> <li>◆ MS Office, Unity, IntelliJ Idea, Wireshark, Cisco Packet Tracer, Tableau, SPSS</li> </ul>	<ul style="list-style-type: none"> <li>◆ Eskisehir Math&amp;Science Competition- Bronze Medal-2007</li> <li>◆ Effective Communication-2013</li> </ul>
EXTRACURRICULAR ACTIVITIES	<ul style="list-style-type: none"> <li>◆ Database: Oracle PL/SQL, MySQL Workbench, PHP MyAdmin,SAS</li> <li>◆ Programming: Java, C, C#, C++, Visual Basic, Html, PHP, CSS, JavaScript, Python, R.</li> <li>◆ Systems: Windows 7/8/10, UNIX</li> <li>◆ Hardware: VHDL, Logic Gates, Arduino, RFID, Color Detection</li> </ul>	<ul style="list-style-type: none"> <li>◆ Diction and Body Language-2013</li> </ul>
		LANGUAGES
PERSONAL INFORMATION	<ul style="list-style-type: none"> <li>◆ Basketball: Kılıcoglu High School Basketball Team player (2007 - 2008)</li> <li>◆ Member, ÖzÜ E-Sport Club (2014 - 2015)</li> <li>◆ Licensed Scuba Diving: First Star Diver(2011-Present)</li> </ul>	<ul style="list-style-type: none"> <li>◆ Written oral fluency in English</li> <li>◆ Pre-Intermediate level French</li> <li>◆ Beginner level Spanish</li> </ul>
	Birth Date: 19/06/1995 Driver's License: B Class	

Figure [55]: Candidate Panel Auto-Fill Function Input Example 2

### Candidate Information Confirmation

Dear Candidate you can see registered information of yours, after the auto fill transaction.If there is a incorrect information you can go Candidate Settings Page and Update.

Execution Time : 5.8260359764099

First name : **Güzide**

Last name : **Akarsu**

City : **Kocaeli**

E-mail : **guzideakarsu@gmail.com**

Phone Number : **+905426299228**

Birthday : **1970-01-01**

Foreign Languages:

**English**

Experience:

**HR Analytics Manager at Vodafone**

**Manager- Marketing / Data Analytics at Garanti Payment System**

**Specialist-Risk Strategies at Denizbank**

Skills:

**Voice of the Customer**

**customer journey mapping**

**CRM**

Driving Licence: No

Disability: No

Master Education: **istanbul university**

Marital Status: divorced

University: **MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ**

Figure [56]: Candidate Panel Auto-Fill Function Output Example 3



### Candidate Settings

In this section, you can see registered information of yours.If there is a incorrect information you can fill the form and submit.

First name:

Last name:

City:

E-mail:

Phone Number:

Birthday:

Foreign Language Number:

Experience:

Skills:

Driving Licence:


Disability:

Master Education:

Marital Status:

University:

Profile Picture



Take a New Photo

Submit

Figure [57]: Candidate Panel Candidate Settings Section

### CANDIDATE ACCEPTANCE STATUS

Dear Candidate your acceptance information is shown below.

When we look at the your applied records: We have '3' saved records.

For job which name is CRM Solutions Senior Specialist

Now your job application is analyzing by our Human Resources team.Thanks for your patience.

For job which name is Billing Services Operations Senior Specialist

Your job application is accepted by our Human Resources team.We will contact you about interview.

For job which name is IP CPN 1st Level Operations Specialist

Now your job application is analyzing by our Human Resources team.Thanks for your patience.

---

Candidate Picker Website

You logged in as a Candidate

Figure [58]: Candidate Panel Acceptance Status Section

```

<?php
// This username password and port information changed on Server
$host="localhost";
$username="root";
$password="1995";
$dbname="candidatedb";
$port="3307";

try {
    $conn = new PDO("mysql:host=$host;port=$port;dbname=$dbname;charset=utf8", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    echo 'PDO Exception: ' . $e->getMessage();
    exit();
}
?>

```

Figure [59]: Database Connection Php Code

```

<div id="page_content">
    <div class="section">

        <h1>Dear Admin</h1>

        <h2>Please enter your login information</h2>
        <form action="adminloginresult.php" method="POST">
            User Name:<br>
            <input type="text" name="username" class="login"><br>
            Password:<br>
            <input type="password" name="password" required class="login"><br><br>
            <input type="submit" name="submit" id="submit" value="Login" class="login"/>
        </form>
        <br>
        <div class="register">
            <a href="registeradmin.php">Do not have any account click to register.</a>
        </div>

        <hr>
    </div>
</div>

```

Figure [60]: Sample Form Creation with Using HTML

```

<div class="navigation">
<div id='cssmenu'>
    <ul>
        <li><a href="candlogin.php">Home</a></li>
        <li><a href="jobposting.php">Job Posting</a></li>
        <li><a href="applyajob.php">Apply for A Job</a></li>
        <li><a href="uploadcv.php">Upload CV</a></li>
        <li><a href="cvphoto.php">Upload CV by taking photo</a></li>
        <li><a href="candidatesetting.php">Candidate Settings</a></li>
        <li><a href="candstatus.php">Candidate Apply Status</a></li>
        <li><a href="logout.php">Log out</a></li>
    </ul>
</div>
</div>

```

Figure [61]: Sample Candidate Menu Bar Creation

```
<?php
$msg = "";
if(isset($_POST['submit'])) {
    $username = trim($_POST['username']);
    $password = trim($_POST['password']);
    if($username != "" && $password != "") {
        try {
            $query = "select * from `users` where `username`=:username and `password`=:password";
            $stmt = $conn->prepare($query);
            $stmt->bindParam('username', $username, PDO::PARAM_STR);
            $stmt->bindValue('password', $password, PDO::PARAM_STR);
            $stmt->execute();
            $count = $stmt->rowCount();
            $row = $stmt->fetch(PDO::FETCH_ASSOC);
            if($count == 1 && !empty($row)) {
                /***** Your code *****/
                session_start();
                $_SESSION['name'] = $username;
                echo 'Succesfully Logged In';
                header("location:mainpage.php");
            } else {
                $msg = "Invalid username and password!";
                echo $msg;
            }
        } catch (PDOException $e) {
            echo "Error : ".$e->getMessage();
        }
    } else {
        $msg = "Both fields are required!";
        echo $msg;
    }
}
?>
```

Figure [62]: Sample Login System Php Code

```
input[type="radio"] {
    height:12px;
    margin-right:-50px;
    margin-left:-15px;
    font:12px Arial;
}
input[type="file"] {
    margin:auto;
    font:12px Arial;
}
.custom-file-upload {
    border: 1px solid #ccc;
    display: inline-block;
    padding: 6px 12px;
    cursor: pointer;
}
input.upload{
    margin:5px;
    width: 27em;
    height: 2em;
    border-radius: 15px;
    border: 1px solid #999;
}
```

Figure [63]: Sample CSS Styling Code

```

<h2>SELECT CANDIDATE</h2>
<p>This section is for view candidate information.You can easily select

<?php
include 'databaseconnection.php';
$query = $conn->prepare("SELECT * FROM `candidates`");
$query->execute();

?>
<form action="selected.php" method="post">
<select name="dynamic_data">
<option value='selected'>
Select a Candidate
</option>
<?php
$i=0;
while($result = $query->fetch(PDO::FETCH_ASSOC)) {
?>
<option value="<?=$result['candidateID'];?>">
<?=$result['firstName'];?>
<?=$result['lastName'];?>
</option>

<?php
        $i++;
    }
?>
</select>
<input type="submit" name="submit" value="Submit"/>
</form>

```

Figure [64]: Sample Form with Using Pdo Fetch Technique

```

$partialState = $_POST['partialState'];
$target = $_POST['target'];
include 'databaseconnection.php';

$lang="SELECT `university`,`firstName`,`lastName`,`candidateID` FROM `candlang` WHERE combinedlang LIKE :like";
$skill="SELECT `university`,`firstName`,`lastName`,`candidateID` FROM `candskills` WHERE combinedskill LIKE :like";
$experience="SELECT `university`,`firstName`,`lastName`,`candidateID` FROM `candexp` WHERE combinedexp LIKE :like";

if($target=="Language"){
    $query = $conn->prepare($lang);
}
else if($target=="Skills"){
    $query = $conn->prepare($skill);
}
else if($target=="Experience"){
    $query = $conn->prepare($experience);
}

$query->execute(array(':like'=>'%'.$_POST['partialState'].'%'));

$result = $query->fetchAll();

foreach($result AS $state){
    echo '<div>'.$state['university'].'</div>';
    echo '<div>'. "Candidate ID: ".$state['candidateID']. "&nbsp;";
    echo "Candidate Name&nbsp;:&nbsp;". $state['firstName']. "\t". $state['lastName'];
    echo " <input type='button' onclick='win(this.id)' id='".$state['candidateID']."' value='Show Candidate' /> ". "</div>";
}

```

Figure [65]: Php Dynamic Search Candidate for Language-Skill-Experience

```

<script src="http://code.jquery.com/jquery-latest.min.js" type="text/javascript"></script>
<script type="text/javascript">
function getStates(value) {
$.post("getpart.php", { "partialState":value },
function(data) {
$("#results").html(data);
});
}
</script>

```

Figure [66]: JavaScript Search Creation Code

```

<script type="text/javascript">
var calculateRank = function () {
$.ajax({
success : function() {
$('#resultdiv').load("dynamicrank.php").fadeIn("slow");
setTimeout(calculateRank, 60000);
},
});
};
calculateRank();
</script>

```

Figure [67]: Candidate Ranking Section Page Update Every Minute Using AJAX

```

$(document).ready(function() {
$(":button").click(function() {
$.css(":button", 'background-color', 'white');
$(this).css('background-color', '#87CEFA');
});
});

```

Figure [68]: Change Button Colours Function Script Side

```

$folderdirectory = "testupload/";
$folderdirectory = $folderdirectory. basename( $_FILES['file']['name'] ) ;
$file_type=$_FILES['file']['type'];
$filename=basename( $_FILES['file']['name']);
if(file_exists($folderdirectory)){
    array_map('unlink', glob($folderdirectory));

    if ($file_type=="application/pdf" || $file_type=="image/jpeg" || $file_type== 'text/txt') {
        if(move_uploaded_file($_FILES['file']['tmp_name'], $folderdirectory))
        {
            $message="The file ". basename( $_FILES['file']['name']). " is uploaded.";
            ?>
            <p>Succesfully added file to the server.</p>
            <p> <?php echo $message; ?></p>
            <p>Dear Candidate if you want to fill your personal information by using uploaded CV Please click Auto Fill button.</p>
            <form action="autofillarea.php" method="POST">
                <input type="hidden" name="fName" value="<?php echo $_FILES['file']['name'];?>"><br>
                <input type="submit" name="submit" value="Auto Fill" ,/>
            </form>
            <?php
        }
        else
        {
            ?> <p>Cv upload is not complete.</p> <?php
            echo "Problem occured while uploading file";
        }
    }
    else {
        ?> <p>Cv upload is not complete.</p> <?php
        echo "You may only upload PDFs or JPEGs files.<br>";
    }
}

```

Figure [68]: Upload File Section Checking File Constraints and Passing File Name to the Auto-Fill Function

```

<?php
$fileName = $_POST['fName'];

$time_start = microtime(true);
require_once("Java.inc");
$javabridge = new java("javabridgedemo.javaBridge");
$ocrPart = new java("javabridgedemo.objectCharacterRecognition");
$folderRead = new java("javabridgedemo.folderRead");

$ocrPart->ocrSide($fileName);
$docname=substr($fileName,0,-3);
$docname=$docname.".txt";
$folderRead->filesInTheFolder($docname,$candId);
?>
<p>
<?php echo "Execution Time : ".$execution_time = (microtime(true) - $time_start);?></p>

```

Figure [69]: Php-Java Connection Setup (Accessing Java Functions via PHP)

```

else if( arr[iy].contains("phone") || arr[iy].contains("+90") || arr[iy].contains("gsm")){
    String phone = arr[iy];
    String phoneno="";
    String[] splitNo = phone.split("\\s+");
    for(int x=0;x<splitNo.length;x++) {
        System.out.println(splitNo[x]);
        if(splitNo[x].contains("5")) {
            phoneno=splitNo[x];
        }
    }

    try {
        PreparedStatement st = con2.prepareStatement("Update candidates2 Set phone_no = ? where candidateID = ? ");

        st.setString(1, phoneno);
        st.setString(2, user);
        st.executeUpdate();
    } catch (SQLException e) {

        e.printStackTrace();
    }
}
}

```

Figure [70]: Simple Example of The Keyword Search Algorithm

```

<script language="JavaScript">
    function take_snapshot() {

        Webcam.snap( function(data_uri) {

            Webcam.upload( data_uri, 'saveimageforusers.php', function(code, text) {
                document.getElementById('results').innerHTML =
                '';
            } );
        } );
    }

    $(document).ready(function() {
        $("#results").hide();
        $("#hide").click(function() {
            $("#results").show();
        });
    });
</script>

```

Figure [71]: Candidate Profile Picture Take a Photo Using Webcam

```
<?php
    session_start();

    $candId=$_SESSION['name'];
    $jobId=$_POST['selectedjob'];
    $desc=$_POST['jobReason'];
    $jobpoint=0;

    include 'databaseconnection.php';
    try{
        $query = $conn->prepare("INSERT INTO Job_enrollment values('$candId','$jobId','$jobpoint','$jobapplyreason')");
        $res = $query->execute();

    }catch (PDOException $e) {
        echo 'You already apply this job which Name is'.$jobName.' '
        echo 'PDO Exception: ' . $e->getMessage();
        exit();
    }

    $query2 = $conn->prepare("SELECT * FROM `jobs` where jobId='$jobId'");
    $query2->execute();
    $result = $query2->fetch(PDO::FETCH_ASSOC);
    $jobName = $result['jobName'];

    ?>

    <p> You succesfully apply job which Name is <?php echo $jobName.' ' ?> </p>
```

Figure [71]: Apply a Job Section Sample Code

```
<?php

    require 'mail/phpmailer/PHPMailerAutoload.php';
    $mail= new PHPMailer;
    $mail->isSMTP();
    $mail->Host='smtp.gmail.com';
    $mail->Port=587;
    $mail->SMTPAuth=true;
    $mail->SMTPSecure='tls';

    $mail->Username='candidatepicker@gmail.com';
    $mail->Password='enes2005';

    $mail->setFrom('candidatepicker@gmail.com','Hiring Process Application');
    $mail->addAddress($email);
    $mail->addReplyTo('candidatepicker@gmail.com');

    $mail->isHTML(true);
    $mail->Subject='Apply Job Information Message';
    $str='Dear '.$name.'';
    $mail->Body=$str."<br><p> You succesfully apply job</p>";

    if(!$mail->send()){
        ?>
        <p>Information Mail can not sent to your registered mail address</p>
        <?php
        }
        else{
            ?>
            <p>Information Mail sent to your registered mail address</p>
            <?php
        }
    }

    ?>
```



Figure [70]: PHP Mailer Class Sample Usage Code

```
<script>
function myFunction(val) {
    var i=1;

    if(val==1){
        my_div.innerHTML=" ";
        my_div.innerHTML = my_div.innerHTML +i+".Language:<br><input type='text' name='mytext'>";
        addOption();
    }
    else if(val==2){
        my_div.innerHTML=" ";
        my_div.innerHTML = my_div.innerHTML +i+".Language:<br><input type='text' name='mytext'>"
        i++;
        my_div.innerHTML = my_div.innerHTML +i+".Language:<br><input type='text' name='mytext+i'>"
        addOption();
    }
}
```

Figure [71]: Creating Language Labels with JavaScript Part of The Code

```
<h2>CANDIDATE ACCEPTANCE STATUS</h2>
<p>Dear Candidate your acceptance information is shown below.</p>
<?php
session_start();
$candId=$_SESSION['name'];

include 'databaseconnection.php';
$query = $conn->prepare("SELECT * FROM `job_enrollment` WHERE cand_id='$candId'");
$query->execute();

$result = $query->fetch(PDO::FETCH_ASSOC);
$status = $result["acceptance_status"];
$status=strtolower($status);
if($status=="analyzing"){
    ?>
    <p>Now your job application is analyzing by our Human Resources team.Thanks for your patience.</p>
    <?php
}else if($status=="accepted"){
    ?>
    <p>Your job application is accepted by our Human Resources team.We will contact you about interview.</p>
    <?php
}else if($status=="rejected"){
    ?>
    <p>Your job application is rejected.Good luck for your job search.</p>
    <?php
}
?>
```

Figure [72]: Candidate Acceptance Status View Code

```

<?php
include 'databaseconnection.php';
session_start();

$id=$_POST['id'];
$status=$_POST['status'];
$changedstatus=$_POST['changed'];

$candid=$_SESSION['candid'];

try{
$query = $conn->prepare("Update `job_enrollment` set acceptance_status='$changedstatus' where cand_id='$candid' and job_id='$id'");
$res = $query->execute();

?>
<p>Succesfully changed current candidate status
"<?php echo $status; ?> " to " <?php echo $changedstatus; ?> "
for candidate id <?php echo $candid; ?>.</p>

<?php

}catch (PDOException $e) {
    echo 'Failed.';
    echo 'PDO Exception: ' . $e->getMessage();
    exit();
}

?>

```

Figure [73]: Admin Update Candidate Acceptance Status Code

```

$query2 = $conn->prepare("SELECT * FROM `candidates2`");
$query2->execute();
while($result2 = $query2->fetch(PDO::FETCH_ASSOC)) {
    $candpoint="0";
    $id = $result2["candidateID"];
    $lname = $result2["lastName"];
    $languagenum = $result2["languagenum"];
    $dlicenc = $result2["dlicenc"];
    $maritalstat = $result2["maritalstat"];
    $master = $result2["master_educ"];
    $disable = $result2["disability"];
    $age = $result2["age"];
    $candpoint = $result2["candpoint"];
    if($languagenum >= "2"){
        $candpoint=$candpoint+$langpo;
    }
    if($dlicenc == 1){
        $candpoint=$candpoint+$licpo;
    }
    if($disable == 1){
        $candpoint=$candpoint+$disablepo;
    }
    if($master != "null" ){
        $candpoint=$candpoint+$masterpo;
    }
    if($maritalstat == "single"){
        $candpoint=$candpoint+$maritalpo;
    }
    if($age <= "25"){
        $candpoint=$candpoint+$youngpo;
    }
}
$query3 = $conn->prepare("Update `candidates2` set candpoint='$candpoint' where candidateID='$id'");
$query3->execute();
$candpoint="0";
}

```

Figure [74]: Setting Automatic Point System Code

```

@Test
public void breachTesting() {
    System.setProperty("webdriver.edge.driver", "C:\\Users\\Hp\\Desktop\\selenium\\MicrosoftWebDriver.exe");
    WebDriver webDriver = new EdgeDriver();
    webDriver.get("http://localhost/mainpage.php");

    //User try to enter admin main page with entering URL
    //Session Control detects it and direct to the login page.
    String pageURL = webDriver.getCurrentUrl();
    assertEquals(pageURL, actual: "http://localhost/adminlogin.php");
}

```

Figure [75]: Login Breach Testing

```

@Test
public boolean searchCandidate() {
    System.setProperty("webdriver.edge.driver", "C:\\Users\\Hp\\Desktop\\selenium\\MicrosoftWebDriver.exe");
    WebDriver webDriver = new EdgeDriver();
    webDriver.get("http://localhost/searchcandidate.php");
    WebDriverWait wait = new WebDriverWait(webDriver, timeOutInSeconds: 10);
    //Breach Prevention is disabled to test faster
    WebElement cityButton = webDriver.findElement(By.id("citybutton"));
    cityButton.click();

    WebElement searchBox = webDriver.findElement(By.className("searchset"));
    searchBox.click();
    searchBox.sendKeys(...charSequences: "esk");
    try {
        wait.until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//*[@contains(text(), 'eskisehir')]")));
        return true;
    } catch (Exception e) {
        return false;
    }
}

```

Figure [76]: Search Candidate City Testing

```

@Test
public void searchButtonColor() {
    System.setProperty("webdriver.edge.driver", "C:\\Users\\Hp\\Desktop\\selenium\\MicrosoftWebDriver.exe");
    WebDriver webDriver = new EdgeDriver();
    webDriver.get("http://localhost/searchcandidate.php");
    //Breach Prevention is disabled to test faster
    WebElement universityButton = webDriver.findElement(By.id("unibutton"));
    universityButton.click();
    String buttonColor = universityButton.getCssValue(S: "background-color");
    //This '#87CEFA' value stands for color light blue.
    assertEquals(buttonColor, actual: "#87CEFA");
}

```

Figure [77]: Candidate Search Button Style Testing