Online Recruitment System with Advanced Search Capabilities

A dissertation submitted to The University of Manchester for the degree of MSc e-Business Technology in The School of Informatics

2007

Yacoub Sabatin

School of Informatics
# List of Contents:

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of Contents</td>
<td>2</td>
</tr>
<tr>
<td>List of Figures</td>
<td>5</td>
</tr>
<tr>
<td>List of Tables</td>
<td>6</td>
</tr>
<tr>
<td>Abstract</td>
<td>7</td>
</tr>
<tr>
<td>Declaration</td>
<td>8</td>
</tr>
<tr>
<td>Copyright Statement</td>
<td>9</td>
</tr>
<tr>
<td>Dedication</td>
<td>10</td>
</tr>
<tr>
<td>1 Introduction</td>
<td>11</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>11</td>
</tr>
<tr>
<td>1.2 Project Description</td>
<td>11</td>
</tr>
<tr>
<td>1.3 Contribution</td>
<td>12</td>
</tr>
<tr>
<td>1.3.1 Instant search</td>
<td>12</td>
</tr>
<tr>
<td>1.3.2 Ranking technique</td>
<td>13</td>
</tr>
<tr>
<td>1.4 Dissertation organization</td>
<td>16</td>
</tr>
<tr>
<td>1.4.1 Chapter 1: Introduction</td>
<td>16</td>
</tr>
<tr>
<td>1.4.2 Chapter 2: Background</td>
<td>16</td>
</tr>
<tr>
<td>1.4.3 Chapter 3: Success Criteria</td>
<td>17</td>
</tr>
<tr>
<td>1.4.4 Chapter 4: Design</td>
<td>17</td>
</tr>
<tr>
<td>1.4.5 Chapter 5: Realisation</td>
<td>17</td>
</tr>
<tr>
<td>1.4.6 Chapter 6: Evaluation and Testing</td>
<td>17</td>
</tr>
<tr>
<td>1.4.7 Chapter 7: Conclusions and Further Work</td>
<td>17</td>
</tr>
<tr>
<td>2 Background</td>
<td>18</td>
</tr>
<tr>
<td>2.1 Overview</td>
<td>18</td>
</tr>
<tr>
<td>2.2 Related works</td>
<td>18</td>
</tr>
<tr>
<td>2.2.1 Monster.com</td>
<td>18</td>
</tr>
<tr>
<td>2.2.2 HotJobs</td>
<td>19</td>
</tr>
<tr>
<td>2.2.3 Salary.com and similar examples</td>
<td>19</td>
</tr>
<tr>
<td>2.2.4 Adecco</td>
<td>19</td>
</tr>
<tr>
<td>2.2.5 GetTheJob.com</td>
<td>20</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>2.2.6 Specialized job portals</td>
<td>20</td>
</tr>
<tr>
<td>2.3 Summary of related works</td>
<td>21</td>
</tr>
<tr>
<td>3 Success Criteria</td>
<td>23</td>
</tr>
<tr>
<td>3.1 Method of investigation</td>
<td>23</td>
</tr>
<tr>
<td>3.2 How to determine whether the success has been achieved.</td>
<td>26</td>
</tr>
<tr>
<td>4 Design</td>
<td>27</td>
</tr>
<tr>
<td>4.1 Conceptual design</td>
<td>27</td>
</tr>
<tr>
<td>4.1.1 Database and ER diagram</td>
<td>27</td>
</tr>
<tr>
<td>4.1.2 Site map</td>
<td>27</td>
</tr>
<tr>
<td>4.2 Web application design</td>
<td>28</td>
</tr>
<tr>
<td>4.2.1 Three layers design</td>
<td>28</td>
</tr>
<tr>
<td>4.2.2 Other design issues</td>
<td>29</td>
</tr>
<tr>
<td>4.3 The vacancies ranking equation</td>
<td>30</td>
</tr>
<tr>
<td>4.4 How will the design meet the success criteria?</td>
<td>30</td>
</tr>
<tr>
<td>5 Realisation</td>
<td>31</td>
</tr>
<tr>
<td>5.1 System specifications and description</td>
<td>31</td>
</tr>
<tr>
<td>5.1.1 The project</td>
<td>31</td>
</tr>
<tr>
<td>5.1.2 Database tables and design</td>
<td>33</td>
</tr>
<tr>
<td>5.1.2.1 Tables</td>
<td>33</td>
</tr>
<tr>
<td>5.1.2.2 Relations in the Database</td>
<td>37</td>
</tr>
<tr>
<td>5.2 Scenario of the matching</td>
<td>37</td>
</tr>
<tr>
<td>5.3 Features and system characteristics</td>
<td>39</td>
</tr>
<tr>
<td>5.4 Tools and technologies used</td>
<td>39</td>
</tr>
<tr>
<td>5.5 Pages of the Web Application and notes about the</td>
<td>41</td>
</tr>
<tr>
<td>Implementation</td>
<td>41</td>
</tr>
<tr>
<td>5.5.1 Notes about the Implementation</td>
<td>41</td>
</tr>
<tr>
<td>5.5.2 Main Pages and functions of the Application</td>
<td>44</td>
</tr>
<tr>
<td>6 Evaluation/Testing</td>
<td>57</td>
</tr>
<tr>
<td>6.1 Testing and evaluation</td>
<td>57</td>
</tr>
</tbody>
</table>
List of Figures:

Figure 1: Portal pattern .......................................................... 11
Figure 2: Locations of the users who did the survey ................. 24
Figure 3: Basic relations between data entities.......................... 27
Figure 4: Site map ................................................................. 28
Figure 5: The separation of concerns in development ................... 29
Figure 6: Relations among tables .............................................. 37
Figure 7: Flowchart: Scenario of matching for job seeker ............ 38
Figure 8: Flowchart: Scenario of matching for employers .......... 38
Figure 9: Configure the local server to send emails .................... 44
Figure 10: Home page .......................................................... 45
Figure 11: Updating employer profile ....................................... 47
Figure 12: Adding new vacancy .............................................. 48
Figure 13: Showing and printing CVs ....................................... 49
Figure 14: Browsing vacancies by category .............................. 50
Figure 15: An employer receiving a notification by email ............. 51
Figure 16: Browsing job seekers details in a category ................ 51
Figure 17: Listing ordered relevant vacancies to the job seeker ....... 52
Figure 18: Examples of custom search done by a job seeker ......... 54
Figure 19: Project plan ......................................................... 79
List of Tables:
Table 1: General characteristics of investigated online job portals..............21
Table 2: General characteristics of the proposed job portal .........................39
Table 3: Survey users' responses ..................................................................63
Table 4: Summary of answers ratings ............................................................64
Abstract

Online dedicated portals are popular nowadays; people use them to get services and information all around the world regardless of their physical location and around the clock. Today's emerging technology and tight labour market have incited a stampede to electronic recruiting. Job boards and other kinds of electronic recruitment systems were a notable improvement over newspaper classified ads in terms of timeliness, reach, and cost, however when looking to the current systems, we can see a lot of advanced features, but without being automatic in suggesting relevant answers to both job seekers and employers. In this project we demonstrate a prototype solution in which we built a dedicated website for job seekers, students, or graduates on one hand, and potential employers on the other, the aim is to enable them reach and match each other's requirements using an advanced instant custom and automatic search feature. Both can have access to the website which is introduced in the form of online portal holding their details in a database and allowing searches to match seekers with providers and vice-versa. Our proposed system can be considered as a complex adaptive system (CAS) as the blocks of profiles data are growing non-linearly, as well as the access and searches to these components.

Declaration

No portion of the work referred to in the dissertation has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.
Copyright Statement

(1) Copyright in text of this dissertation rests with the author. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the author. Details may be obtained from the appropriate Graduate Office. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the author.

(2) The ownership of any intellectual property rights which may be described in this dissertation is vested in the University of Manchester, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the University, which will prescribe the terms and conditions of any such agreement.

(3) Further information on the conditions under which disclosures and exploitation may take place is available from the Head of the School of Informatics.
Dedication

"Say: Lo! My worship and my sacrifice and my living and my dying are for Allah, Lord of the Worlds"

Quran 6:162

This work is dedicated to my mother, my father, my brother, and my sisters, for all the love and support they have given me through the years.

It is also dedicated to my teachers and supervisors who taught and helped me in both of Al-Quds University (Palestine) in my first degree, and in The University of Manchester (UK) in my second degree.
Chapter 1

Introduction

An online profile management system has been built to match and put in touch job seekers with their respective employers; in addition to that, a search technique has been developed to enable users to get results based on their search criteria in the most efficient possible order.

1.1 Motivation

When looking to the available online recruitment systems and job portal, we can see the advanced features they provide, however, the search approaches are still traditional, and that is what motivated us to include an automatic search function, with relevancy ranking technique.

1.2 Project Description

To enable users get the results they want, a matching technique was applied, and this was based on integrating some existing widespread standards and classification terms used in a formal CV for job seekers, and similar terms for job posting.

Figure 1: Portal pattern [22]
The project is based on portal composite pattern, i.e.
combine business patterns and integration patterns to create
complex, advanced e-business applications [22] see figure 1.

The system is available on http://www.Alooza.com

Our project system is profile-based rather than resume-based, which is better [32], and more accurate, this means that this model attracts more job seekers, we can learn from the case of Hire.com, whose technology tools were based on job seekers profiles, and since its launch, the initiative has captured approximately one hundred thirty thousand profiles, roughly ten times the resume-based traffic; data entry or resume scanning has been reduced by 80 percent; recruiting cycle time has been decreased 20 percent [32].

1.3 Contribution

In this project, we introduce two different features. Mainly, the instant search which allows the job seeker to get a list of available vacancies listed according to certain criteria, and allows the employers to get a list of potential job seekers instantly, the second thing is the ranking technique itself that orders those vacancies.

1.3.1 Instant search

The job search is a sales and marketing endeavour. The job seeker is the product and the salesperson in the same time, it is important for the job seeker to be able to define her/his customers and promote her/himself to them[37].

The highly successful salesperson knows exactly the product, but to be able to deliver this message to potential customers (the prospective employers in this case), within a very large number of employers, it is important to have an

1 I called the system ‘Alooza’, actually, this word does not mean anything, it is not even in the dictionary, but this is the first word I thought about, when I tried to register a domain name for a project, and I intended to use a word that has no meaning to make it unique.
automatic technique to search for them within the system database depending on the features of the product mentioned before (the job seeker skills) in fastest possible way, that is why the application looks for relevant possible answers for the users as soon as they enter their information, for example, when the job seeker enters her/his profile information and preferences, the system directly provide her/him with relevant vacancies from the database of employers, the same thing happened when an employer add a new vacancy, in this case the system suggests a list of potential job seekers. This technique is not available in job portals we tested in chapter 2.

1.3.2 Ranking technique

In real life, the database will have a huge number of records, whether for vacancies available for job seekers, or for job seekers themselves, and because of this, it is very important to give large number of results to the users in useful way, basically to be ordered in a logical way, i.e. there is no point if the results shown are randomly listed, to solve this issue, an equation was developed to give a rank for each vacancy according to its relevancy with the corresponding job seeker.

To build this equation or ranking algorithm for vacancy rank (VR), we had to find how different fields of information and data are important to users.

**Job Category (a):**
There are many reasons why employers will want to employ somebody, the first reason may include her/his relevant experience [31], employers prefer to recruit a person who have experience in a similar industry as theirs, on the other hand, job seekers prefer to work in a familiar industry most the time, because of this; we gave 7 points for the similarity between the vacancy category (Vacancies: CategoryID)² and the most recent job category for the job seeker (JobSeekers: JobCategory).

---
² TableName:FieldName in our database, for more information about database structure, see chapter 4 and 5.
Type of the job (b):
This is a very important factor, as the job seeker prefers or wants a particular type, in the same time, the employer offers particular vacancy, for example, if the vacancy is available in 'part time' mode, it should not be so relevant for a job seeker who is seeking a full time job.

Sometimes, the job seeker may realize that she/he is in a dead-end job and there is nothing left to challenge her/him, this mostly happens if the job is short-term work assignment, or a contract in a project, in general, with time-defined project work [33], which means that the type of job can be a key factor in choosing a vacancy according to the very customized requirements of the job seeker; that is why we gave this element 4.7 points, in our database (Vacancies: Type) with (JobSeekers: PreferredWork).

Position (c):
This is also a previous-experience-factor, an we gave it 5 points, for the relevancy between (Vacancies: Position) and (JobSeekers: MostRecentJobTitle) in our database, it is one of the most important fields to consider in our equation, many authors [37] emphasized on this as one of the first things to look for, and that is why we gave it the 5 points.

Number of experience years required/preferred (d):
The employer prefers to get a candidate with number of experience years equals to the required number of experience years for a certain vacancy or more, however the job seeker prefers to get a vacancy relevant to her/him, but not too far neither too narrow with respect to her/his experience, that is why we gave 3 points for equality of these two elements and only 2 points if the job seeker experience is more than the required, the two elements here are: (Vacancies: RequiredExperience) and (JobSeekers: ExperienceLevel).

Relevant qualifications (e):
Relevant qualifications becomes in the next position after experience according to [31] and before other personal attributes, and because of this we
gave 3.5 points for the relevancy between (Vacancies: PreferredEducationLevel) and (JobSeekers: EducationLevel).

**Availability period:**
This is the most important element, as there is no point of proposing a vacancy to a job seeker when the employer does not offer this vacancy at the moment, and because of this we did not include this element in the equation explicitly, but we took it into consideration in the stage of testing the vacancies, i.e. the system will only show (and calculate VR for) the vacancies that are available now, in other words when the current date is between the fields (AvailableFrom) and (AvailableTo).

**City and Country (g) and (h):**
The place of work is important, and we have included two fields in each job seeker profile, for the preferred city and country of work, some users will just ignore these if they are not interested in a certain country or city, but some may be interested in these criteria, for the relevancy between (Vacancies: Addresses: City) and (JobSeekers: PreferredCityOfWork) we gave 3.5 points. And for the relevancy between (Vacancies: Addresses: Country) and (JobSeekers: PreferredCountryOfWork) we gave 4.7 points, as it is more difficult to move between countries rather than cities.

**Salary (i):**
Some authors [34] put the salary preferences and good pay in the top, however many others [35][36] emphasize that salary (especially the base salary) should not be a key element, and it is negotiable all the time, and because of this we only gave it 2.5 points when (Vacancies: Salary) is greater than or equal to (JobSeekers: SalaryPreference).

**Preferred job category 1 and 2 (j) and (k):**
When the job seeker specifies a preferred industry or job category to work in, we have to take this into consideration, but with less importance than the previous job category as it is more relevant to the new job, that is why we gave 3.5 points for the relevancy between (Vacancies: CategoryID) and
(JobSeekers: PreferredJobCategory1), and a little less of 2.6 points for the relevancy between (Vacancies: CategoryID) and (JobSeekers: PreferredJobCategory2).

And by aggregating the previous elements we get:

\[
VR_1 = 7a + 4.7b + 5c + (3 \text{ or } 2)d + 3.5e + 3.5g + 4.7h + 2.5i + 3.5j + 2.6k \quad \ldots(1)
\]

For a maximum rank (full relevancy), \(a=b=c=d=e=f=g=h=i=j=k=1\):

\[
VR_2 = 7 + 4.7 + 5 + 3.5 + 3.5 + 4.7 + 2.5 + 3.5 + 2.6 = 40 \quad \ldots(2)
\]

And to get a percentage, multiply \(VR_2\) by 100/40, we get equation (3):

\[
VR = (7a + 4.7b + 5c + (3 \text{ or } 2)d + 3.5e + 3.5g + 4.7h + 2.5i + 3.5j + 2.6k) \times 100/40
\]

The last equation (3) was adopted as the vacancy rank equation, i.e. VR is calculated for each possible vacancy, then listed according to VR.

### 1.4 Dissertation organization

In addition to this chapter, this dissertation includes six more chapters. Our contributions is described in this chapter, and mainly implemented in chapter 5. In the following, we summarize the main contents of the different chapters:

#### 1.4.1 Chapter 1: Introduction

This chapter gives an introduction on the work included in this dissertation; this mainly includes an overview on the project, listing of our contributions, and the structure of this dissertation.

#### 1.4.2 Chapter 2: Background

This chapter includes a review of number of the most popular job portal, their features, and limitations.
1.4.3 Chapter 3: Success Criteria

This chapter gives details about the criteria we set to test the success of the project, along with how we will test these criteria.

1.4.4 Chapter 4: Design

In this chapter, we introduce the database design, site structure, and the methods of interface design.

1.4.5 Chapter 5: Realisation

This chapter introduces the actual implementation of the project, starting with the project description, requirements, database, notes about implementation, and the main features of the system.

1.4.6 Chapter 6: Evaluation and Testing

In this chapter, we introduce the steps we did to test the system success, along with how this has been done, we also include the users' feedbacks and data, and analyse them.

1.4.7 Chapter 7: Conclusions and Further Work

In this chapter, we conclude by the features of our work with the expectation from the future works.
Chapter 2:

Background

2.1 Overview

Online recruitment is one of the most active ecommerce areas [39]; the growth of web recruitment has been driven by a combination of actual costs savings in the recruitment process, increased ease and efficiency for the employer along with an improved experience for candidates.

For job seekers, a recent research [25] shows that 69.0% of job seekers see that Internet has been 'very helpful' in their job search, and 22.3% see it 'somewhat helpful', that is a sum of 91.3%.

2.2 Related works

There are many job portals available online; the dominant players currently appear to be Monster.com, Salary.com, Recruitsoft, RecruitUSA, Adecco, Brainbench, and a few others, each in its own niche [13].

Here we will demonstrate some examples, and show their principles of work, and investigate the problems, weaknesses, and even features associated with them.

2.2.1 Monster.com

Monster.com is one of the Internet's earliest dotcom domain registrants and the first commercial online job placement destination [5], it is one of the World's leading career networks, the idea is based on posting CV by the user then the system makes it available for employers in the same industry, or business.

Monster offers employers cost-effective and efficient recruiting solutions including real-time job postings, complete company profiles, and résumé searching [11].
One of the innovative ideas in Monster is "Talent Market" [12] where the job seeker can post her/his skills and talents and let potential employers bid on her/him.

### 2.2.2 HotJobs

HotJobs is another example; HotJobs was to initially have been acquired by Monster.com, however, Yahoo's $436 million counteroffer eventually won [5] which shows the importance and value of this online business model. The approach of HotJobs is similar, which depends on CV being posted by the user, after HotJobs became part of Yahoo! Network, Yahoo! very large users profiles was integrated somehow in HotJobs database [6] which gives more advantages for Yahoo! This business model is mainly based on advertising, as it is a heavy traffic online community place, it is also supported by some other products and services which are provided for a fee [14] especially posting a job in the case of Monster.com.

### 2.2.3 Salary.com and similar examples

Salary.com is a popular player in the field; it gets some of jobs data from other portals like Dice, Monster, Net-Temp, and HotJobs. The business is based on advertising, while it also provides some paid services such as Personal Salary Report for job seeker, which is based on the job, location, personal skills, and other company factors [17]. Salary.com enables the basic search by job title, location, or the salary, in general, it focuses on providing detailed salary information organized by job type and by geographic location, jobstar.org, and wageweb.com are similar examples [18][19].

### 2.2.4 Adecco

Adecco is another provider of human resource solutions, these solutions include services such as staffing, outsourcing, permanent recruitment, outplacement and career services, training and consulting [23], with offices
and portals for many countries all around the world, what matters here for us is how their online recruitment approach does work, which is not too different than the previous examples; basically, a search is initiated by the user using the job title or/and location, then they can apply by filling a form online which in turn will be send to the employer. This company works as a HR mediator between companies and job seekers, and their website contains different sections for permanent and short time jobs.

2.2.5 GetTheJob.com

Not all employment sites are portals, GetTheJob.com is a vertical search engine or aggregator that gathers job postings directly from tens of thousands of company websites and organizes them into a searchable database [20]; meaning there are no postings are needed from head-hunters, or third party recruiters, the search is available by the job description or the location, job seekers can browse the vacancies by employer as well, but then, it just sends the user to the employer website, the user can create an account and enter her/his profile there [21]. According to International Association of Employment Websites, the database of jobs represents the hidden job market, as 70% of the postings can not be found on commercial job boards [20], this model is heavily supported by advertisements and PPC (Pay Per Click on job posts) and this money comes from the job providers.

2.2.6 Specialized job portals

The previous examples are general job portals or job search engines, however, there are specialized portals which focus on a certain area, a popular example is Dice.com owned by EarthWeb Network which is considered as the largest online IT jobs portal, with 175,000 high-tech listings [7], this job search engine is specialized in listing IT careers all around the world.

The idea in Dice.com is simple, the user can search for a job, then submit a CV with her/his contact details for that particular job vacancy [8], so this approach depends on human decision on a particular job whether it is suitable
or not, then another 'human' decision comes from the employer after studying that CV. Again, this business model is based on advertising.

### 2.3 Summary of related works

By looking to these online services we can see that they provide some additional services and tools such as résumé builder, which makes it easy for the job seeker to build a professional CV, or covering letter, they also offer services like email notifications on certain jobs.

The problem in job boards is that both employers and job seekers are already growing dissatisfied with them, job boards like CareerMosaic and Monster.com are picked over and require companies to take fast action to beat the competition. Moreover, job boards do not surface passive candidates—those who are employed and not actively seeking a job change [15].

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiative</td>
<td>The process begins when a user, mainly a job seeker, initiate a search using a keyword represents the job title most of the times.</td>
</tr>
<tr>
<td>Matching</td>
<td>When the user gets results for her/his search they can choose the vacancy to apply (human decision).</td>
</tr>
<tr>
<td>Application</td>
<td>The user submits her/his CV as a document in some cases or by filling an application form in other cases, immediately or using a saved copy in her/his profile.</td>
</tr>
<tr>
<td>Business model</td>
<td>The business is mostly supported by advertisements, or by service fees including paid listings in some cases.</td>
</tr>
</tbody>
</table>

Table 1: General characteristics of investigated online job portals

While looking for online jobs and careers portal and services, we found that some of them are general and some are specific for certain location, industry, or even specialized in jobs opportunities for a certain group of people as in CollegeGrad which intends to provide entry level jobs and internships for
college students and recent graduates [24], but in general, we can identify some characteristics in the previous examples, and these characteristics are listed in table 1 above.
Chapter 3

Success Criteria

Because a website can easily be used by millions of users in a single day, any potential usability problems can be damaging. Thus, evaluating the usability of websites is particularly important to e-businesses [40]. Some commonly used methods for testing and evaluating the usability of a website, are inspection based, whereas others are user-based.

To evaluate the system and our different proposed functions, we consider standard criteria for their evaluation. These criteria are mainly based on users' feedback, i.e. user-based, as this is the way to test if a computer system is successful, in both of the user interface and functionality areas.

3.1 Method of investigation

The evaluation was undertaken through a test and a survey, done by number of users (mainly job seekers), and to achieve that we added a number of different jobs/vacancies (30+ jobs) in different employers accounts, and business categories into the database, these vacancies have been gotten from work agencies Websites, Careers Service in the University of Manchester Website, and some were just added randomly.

After that, a random target consisting of number of students from the University of Manchester, friends, and other users (30+ persons) have been approached by email and asked to register in the system as job seekers (who reflect the opinion of other users, as job seekers are the main type of the system users), they have been asked to go through different functions available for them, users had to use the system for the first time, even though a first impression may be an emotional reaction heavily influenced by visuals or environment considerations, this is also useful, since if users coming to a
site have a very positive or negative first impression, it could certainly affect their desire to go further [16].

As the Web system is intended for global use, by users from different countries, website’s content and usability have to be designed from a global perspective and be responsive to cultural sensitivity in language along with appropriate use of color, and presentation [41], and because of this we chose our testers from different parts of the world; UK, United States, Middle East, Africa, and Asia, see figure 2 for the locations of these users.

![Figure 2: Locations of the users who did the survey](image)

After that, a survey was distributed to get those users' answers and evaluations, which shaped our success criteria, this feedback and evaluation of the usability, included a test for the results given by the system, and the accuracy of the matches suggested by the system comparing to the users expectations.

This survey included both subjective and objective questions aiming to test both functionality and usability of the system in a user point of view, and it was conducted online using an electronic form.

In survey, some questions were taken from prepared standard web evaluation tests, for example, the test provided by Powell [16].

---

3 This picture is taken from the survey manager in Survey Gizmo, a Practical Online Survey Software.
Questions of the survey:

• First Impression: [ 1 (poor) - 3 (neutral) – 5 (excellent) ]

• The links and sections organizing in website pages is:
  [ 1 (Too Messy) – 5 (Very well organized) ]
  Notes: _____________________________________________________________

• Functions available for the job seeker:
  [ 1 (not enough at all) – 5 (all of what is needed) ]

• Browse Vacancies categories function is:
  [ 1 (Not useful) – 5 (Very useful) ]

• My CV function is:
  [ 1 (Not useful at all) – 3 (OK) - 5 (Very useful) ]

• How do you evaluate the search vacancies function in general?
  [ 1 (Not useful) – 3 (OK) - 5 (Very useful) ]

• How do you evaluate the results accuracy given by the search vacancies function?
  [ 1 (Irrelevant) – 3 (neutral) 5 (Very accurate) ]

• Do you think the search results were ordered in a way or just random?
  [1(Random), 2(Not ordered), 3(Not ordered correctly), 4(Ordered but not according to my expectations), 5(Ordered according to my needs)]

• Automatic search function is:
  [ 1 (Not useful) – 5 (Very useful) ]

• How do you evaluate automatic search function results:
  [ 1 (Irrelevant) – 3 (OK) - 5 (Very accurate) ]

• How do you evaluate the relevancy of vacancies proposed by the system comparing to your expectations? [ 1 (Poor) – 5 (Excellent) ].
  Notes: _____________________________________________________________

• How do you evaluate the easiness of the system?
  [ 1 (Very difficult) – 3 (OK) - 5 (Very easy) ].

• What is the most you liked about the system? Why?
  ________________________________________________________________

• What is the least you liked about the system? Why?
  ________________________________________________________________
• What other functions would you like to see in the system? Do you have any suggestion? Please write details:
  Notes: ___________________________________________________
  ___________________________________________________

3.2 How to determine whether the success has been achieved

The system will be successful if it delivers the expectations and functions that have been built to the users according to their goals and based on requirements, i.e. to match the job seekers with their perspective employers in a way that is acceptable by users, and because of this, we will assume that the system is successful if it gets 50% positive feedback from those users or more.

The previous survey contains 12 questions in addition to the suggestions question, which means, for the 12 questions the full mark for the system will be 60 for every user, or an average of 30 for each one to consider it as successful according to the 50% positive feedback note mentioned above.

All suggestions and notes in the questions above that we got from the participants were taken into consideration during the improvement period or suggested as future work.
Chapter 4

Design

In this chapter we will introduce the logical units of the system, establish a hierarchy of importance and generality, use the hierarchy to structure relationships among the chunks of data and application functions, and then analyze the functional and aesthetic success of the system.

4.1 Conceptual design

In this section we will introduce the site design and architecture at an abstract level.

4.1.1 Database and ER diagram

The following figure 3 below shows the basic relations between data entities, for details information about how the database was built, refer to chapter 5.

![Diagram](image)

Figure 3: Basic relations between data entities

4.1.2 Site map

The following Figure 4 below shows the site map, by showing the relations between the main pages in the application.
4.2 Web application design

4.2.1 Three layers design

As a Web application project, we adopted the 3 layers design approach [27] which is based on the separation of concerns (see figure 5).

These three Layers of Web development are:

- **Content or Structure**: In our project, this is what the users (visitors) get when they come to the Web page. Content can consist of any kind of data.
XHTML makes up the content layer and it also structures the Web document. In our project we tried to make the separation of the concerns to the maximum, for example, nearly, all of contents are coming from the database which gave us neater design, and easier way to develop without distractions from the design side.

- **Style or Presentation:**
  This determines how the document will look to the user, which is defined by the CSS or styles that indicate how the document should be displayed, and on what media types, the use of CSS enabled us to focus on the implementation and the problem itself rather than to worry about the look and feel of every page, at the same time, this technique allowed to achieve consistency in the design easily.

- **Behavior:**
  This layer is the layer of a Web page that does something. In general this is the programming language, whether it is the client-side script or the server-side script, so it can be anything from JavaScript to the advanced scripting languages like ASP, JSP, PHP, or CGI back-end, in our case this is mainly ASP .NET and in some pages JavaScript interactions on the page is the first level of behavior.

### 4.2.2 Other design issues

Reusability is the likelihood a segment of source code can be used again to add new functionalities with slight or no modification [28] and this concept was
used heavily in the project, for example, the problem of consistent look and feel, and the separation between the back-end and the front-end, was solved by using style sheets (see Appendix B), where each block style can be defined only once and used many times. On the other hand, there were some shared parts that appear in some cases, for example, the functions menu of each kind of users that appears in all of their pages in order to ease the navigation and reduce the curve of learning. For implementing this, a shared public class called 'GlobalFunctions' was created, this class included all of these shared methods in it, this was not only used for interface issue, but also for common functions in the application as a whole, such as connecting to the database to get a value.

Reusable modules and classes reduced implementation time, increased the likelihood that prior testing and use has eliminated bugs and localizes code modifications when a change in implementation is required.

4.3 The vacancies ranking equation

We will use the following equation to order vacancies when listing them to the job seekers, as mentioned in chapter 1:

$$VR=(7a+4.7b+5c+(3\text{ or } 2)d+3.5e+3.5g+4.7h+2.5i+3.5j+2.6k)*100/40$$

4.4 How will the design meet the success criteria?

We will test the system as mentioned in chapter 3, then we will use the users' feedback and notes to improve the system, i.e. we may need to redesign some parts of the system especially on the user interface level.
Chapter 5

Realisation

In this chapter, we will introduce a description of the functional requirements, implementation, and showing how it arises from the design, then we will demonstrate some of the interesting elements of the system, illustrating examples from the output, and some snapshots from the system.

5.1 System specifications and description

5.1.1 The project

The project involved building a web portal for supporting the following stakeholders, with the following functions:

1. Job Seekers:
   a. Register by adding a formal language CV as an application, with full semantic details about the work desired, qualifications, and contact details.
   The system is directly looking for matches in the employers' database (Vacancies), this was implemented by creating a dynamic temporary data structure for all possible matches ordered according to the relevancy (VR mentioned earlier), then this view is shown in a results-page where the user can get more details about each one of these vacancies and choose which of them to apply to or contact, this is more automated way than what have been shown in most of other systems mentioned in the background chapter, a similar method is used for the opposite case.
   b. Search or browse the jobs and profiles of employers available in the system, by country, category, industry or any other field.

2. Employers:
   a. Add the jobs needed, the kind of work wanted, profile, and contact details.
The system is directly looking for matches in the job seekers available in the database upon adding the vacancy by the employer, and lists them.

b. Search or browse for job seekers, by qualifications, type of work desired, category, or any other kind of matching.

Some of the benefits which the employer can gain include noticeably costs savings which can typically be achieved by reducing the size of adverts in conventional print media, gaining occasional candidates for free, having a talent pool of candidates, reduced administration as all administrative work is being done automatically by the system, and pre-filtering candidates as they are already classified in their corresponding categories.

3. System admin:
   a. Get reports about available jobs, job seekers, and the site activities.
   b. Send mass emails to the members of the site or to a certain group of them, this can used for sending newsletters or support messages.
   c. Managing the contents of the system, for example, add adverts, etc.
   d. Add, or edit businesses and jobs categories.

The following categories [6] were adopted (with some modifications based on users' feedbacks) to classify the required or the available jobs and vacancies:

The matching procedure is applied by two ways:

- The system looks for the most relevant possible matches and suggest listing them as soon as the user enters her/his details, this is a faster
approach than just to show them when the user initiate the search function.

- The user can initiate search function, where she/he can specify the data for as more fields as desirable for more accurate result.

Of course this not confined on job seeking user looking for a job, but also for employers looking for an employee.

5.1.2 Database tables and design

5.1.2.1 Tables

The following tables were used:

**Users table:**

- UserID: An auto-number, field for identifying the user, and as a foreign key in relations.
- Email: This is used as a user name for login as well as holding the user email; this is the primary key in this table. It is wanted to use this field as user name, in order to reduce the information the user required to remember, as it is easier to remember the email address, rather than a chosen user name, which will need a separate field as well in that case, this is also a common practice on the Web.
- FirstName: This is used for all kinds of users, whether it is a contact person name in a company in the case of the employer, or the job seeker name.
- LastName.
- UserType: This field holds the type of stake holder, which can be 'A' for admins, 'E' for employers, and 'J' for job seekers, in implementation, the value of this field is stored in a session variable all the time, in order to show the corresponding page(s) for each user account in any time, it is also useful for security concerns as to prevent users from seeing pages that do not belong to them.
- UserPassword.
**Employers table:**
This table stores the employers' data.
- UserID.
- IndustryID: The business or industry category of this employer, this is a foreign key, related to the 'Categories' table which is also used for classifying vacancies.
- Company: The name of the company.
- AddressID: This fields holds the ID of the address of the employer, it is a foreign key related to the 'Addresses' table.
- CompanyType: Direct Employer (D) or Staffing Firm (S).
- NumberOfEmployees
- AverageNumberOfJobOpeningsPerMonth: The previous 2 fields can be useful for the job seekers to get an idea about the company they will work in.

System helps the employers using the previous fields to specify exactly what type of person is required for a certain position; employer can submit more than one vacancy, with an expiry date when the vacancy becomes invalid, see Vacancies table below.

**JobSeekers table:**
- UserID.
- Gender.
- BirthDate.
- AddressID: This is a foreign key related to the 'Addresses' table.
- ResumeLink: A URL for the CV if it is available online, or a video résumé link\(^4\).
- ExperienceLevel: Number of experience years.
- MostRecentEmployer.
- MostRecentJobTitle.
- JobCategory: The category ID of the category of the most recent job.

---

\(^4\) A Video résumé generally refers to a video or a job seeker describing her/his career objective, qualifications, education, and employment history. Historically, this video was distributed via VHS, CD, DVD or CD business card. Today, Video Resumes typically distributed via the Internet using streaming media.
• PreferredJobCategory1.
• PreferredJobCategory2.
• EducationLevel: This can be one of the following [6] Illiterate, Student, Technician, Associates, Bachelor of Science, Bachelor of Business Administration, Bachelor of Arts, Juris Doctorate, Master of Science, Master of Arts, Medical Doctor, Master of Business (MBA), Master of Fine Arts (MFA), PhD, or Other.
• SchoolName: This can hold university or college name.
• GradeOfGraduation.
• LanguageSpoken1.
• LanguageSpoken2.
• LanguageSpoken3.
• SalaryPreference: Minimum preferred salary if applicable.
• PreferredWork: Commitment, i.e. full time, part time, temporary, or any.
• PreferredCityOfWork: If the user prefers to work in particular city.
• PreferredCountryOfWork.
• Reference1.
• Reference1Contact.
• Reference2.
• Reference2Contact.

System can help the user using the previous data as components to generate a formal résumé, the user can include optional data such as hobbies, covering letter text, in a document to be uploaded to the system, and made available to the employers.

**Vacancies table:**

• VacID.
• EmpID: Foreign key related to the 'Employers' table; this field determines which vacancy belongs to which employer.
• CategoryID: The business category of this vacancy, this is a foreign key, related to the 'Categories' table which is also used for classifying employers, notice that it is not necessary to have the same
category/classification for the vacancy and the employer offering this vacancy.

- **Type**: This can be part time with hourly salary, full time with annual salary, or temporary, i.e. a job for a certain period of time for example, in summer.
- **AvailableFrom**: Starting date for listing of this vacancy.
- **AvailableTo**: The closing date of this vacancy, note that employers can use the previous two fields to automate vacancies listings in certain and limited period of time.
- **RequiredExperience**: Number of experience years required/preferred.
- **WorkPlace**: The location of the work, this is a foreign key related to the 'Addresses' table.
- **PreferredEducationLevel**.
- **Salary**: hourly or annual depending on 'Type' field above.
- **Position**.
- **JobDescription**.

**Addresses table:**
This contains the following fields:

- **AddressID**.
- **PrimaryPhone**.
- **SecondaryPhone**.
- **Fax**.
- **Address1**.
- **Address2**.
- **PostalCode**.
- **City**.
- **Country**.
- **WebSite**: This field is included in this shared table as it can be used for both of the job seeker and employer, at the same time it can hold 'more details' link about the job vacancy.
Categories table:

- **CatID**: An auto-number, field for identifying the business category for employers, recent job for job seekers, preferred categories for job seekers, and vacancies.

- **CategoryName**: A proposed set of categories was mentioned above in section 5.1.1, notice that the admin could add/edit/delete categories.

5.1.2.2 Relations in the Database:

The following figure shows tables and relations between them in our system.

![Figure 6: Relations among tables](image)

5.2 Scenario of the matching

The following two figures show the scenarios for both job seekers, and employers.
Figure 7: Flowchart: Scenario of matching for job seeker

Figure 8: Flowchart: Scenario of matching for employers
5.3 Features and system characteristics

In contrast of general systems characteristics mentioned in section 2.3, the followings table 2 lists some of our system features:

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiative</td>
<td>The process can be initiated whenever there is a change in the database, whether it is a profile posting or a job posting, in addition to the traditional search-then-apply method mentioned before.</td>
</tr>
<tr>
<td>Matching</td>
<td>The system will list the possible matches in an order that reflects the relevancy, and then the user can choose which of them to apply to in the case of the job seeker, or which users to contact in case of the employer.</td>
</tr>
<tr>
<td>Application</td>
<td>The formal CV data fields are already there in the user's profile if she/he is a returning user, or it has been entered now if she/he is a new user, this data will be submitted automatically when the user wants to apply. When it is the other case, when an employer is seeking for a candidate, they can contact the job seeker.</td>
</tr>
<tr>
<td>Business model</td>
<td>In real life, such kind of online project can be supported by adverts, subscription fees, and paid services, at the moment we just demonstrate the idea.</td>
</tr>
</tbody>
</table>

Table 2: General characteristics of the proposed job portal

5.4 Tools and technologies used

ASP.NET in Visual Basic .NET flavour was used as the main development language, with Microsoft Access as a database engine; the reason for choosing this combination is the improved performance for ASP.NET pages especially over the previous traditional ASP pages [4], which were always
interpreted at runtime by the script parser. ASP.NET was specifically designed to allow creating applications that would run as effectively on a small intranet as on a large commercial site [3].

ASP.NET also provides code-level security that allows controlling the context in which code runs [3], other features include compiled pages, a caching system for pages already served.

ASP.NET is a very robust development tool, and this can be seen by the fact that a large number of the most demanding Websites in the world run ASP.NET, examples include Microsoft Website itself, Hotmail, MySpace.com, Dell.com, Match.com, Monster.com, Live Search, NHS.com, Shelfari.com, Whooiz.com, LondonStockExchange.com [10], and others.

In general, scalability, security, and performance are the major reasons why we chose ASP.NET

AJAX technology is also used for the purpose of implementing some interactive features in the system where it is intended to make web pages more responsive, and to implement this, we have used .NET AJAX Extensions which has been developed by Microsoft recently as an extension for ASP.NET applications.

Microsoft Visual Web Developer ® 2005 was used as the main development environment, and as a tool to build most of the interfaces. Microsoft Access XP was used for building and exploring the database. Adobe Dreamweaver ® and Microsoft FrontPage ® packages were also used in building some parts of the user interface.

Other tools were also used for the purpose of design, documenting, evaluating and testing.

---

5 For more information about these extension, see [http://www.asp.net/ajax](http://www.asp.net/ajax)
5.5 Pages of the Web Application and notes about the Implementation

In this section we will demonstrate some of the main pages built in this web application, and give tips and notes about important implementation issues that came out.

5.5.1 Notes about the Implementation

**Platform:**
The first thing we have to point to, is the fact that .NET Framework is a full independent platform, which means that we are dealing with a fully object oriented system where all parts are connected together, and the implementation should be carried out in a certain plan, and not randomly, sometimes, this made the life easier, but in some other cases, this gave more time to the planning part of each stage of the implementation process, rather than the actual development.

**Sessions:**
We used Session states variables heavily in the system, and this is normal for a web application where we need to ease the system for the end user and make it more secure.

ASP.NET session state enables us to store and retrieve values for a user as the user navigates the different pages that make up our Web application. The Web server treats each HTTP request for a page as an independent request and that is what is meant by the fact that HTTP is a stateless protocol [29]; by default, the server retains no knowledge of variable values used during previous requests. As a result, building Web applications that need to maintain some cross-request state information can be a challenge.

The answer is coming by ASP.NET session state which identifies requests received from the same user during a limited period of time (determined by 'TimeOut' property) and this is known as a session.

The session provides the ability to persist variable values for the duration of that session.
With the previous Active Server Pages, the Session object was a great place to store user-specific information as users used the application, but there was a problem with this architecture in a Web farm environment [4], however, because the session information was stored in the memory space of the Web server and therefore could not be shared across the different servers, and that is one of the most important changes which have been made to the .NET Framework to support scalability in a Web farm environment.

In our application, all secure users' accounts protected pages use sessions, and sessions' variables.

**Database security:**
As we used MS Access database file to store data, and in addition to normal and manual security procedures, it is also recommended to store it in the designated folder 'App_Data', this helps to keep the database private.

As a new technique, ASP.NET does not allow files in the 'App_Data' folder to be returned if requested directly [30], and this makes the application data more secure, the ASP.NET process identity must be granted read and write file permissions to the Access databases stored in that folder, where the application itself can access data by code.

**Sessions and redirecting pages in the correct way:**
It is a common technique used by many .NET programmers to redirect to a certain protected page after login and creating the required session variables, for example, in C#:

```csharp
Session["Login"] = true;
Response.Redirect("~/Default.aspx");
```

This can work but this is not the right way of doing it [2]

In general, ASP.NET sets a volatile cookie on the client that contains the session token on the creation of a new session, on all subsequent requests, and as long as the server session and the client cookie have not expired.

Redirect sends a special header to the client so that it asks the server for a different page than the one it was waiting for. Server-side Redirect, after sending this header, ends the response. This is a very violent thing to do [2].
What happens really here is that the session token gets lost in the battle; to solve this we had 2 options whether to use the overloaded version of Redirect:

`Response.Redirect("~/default.aspx", false);`

And in the case of the forms authentication, .NET provides a special redirect method for this:

`FormsAuthentication.RedirectFromLoginPage`

According to Microsoft, this method will return the user to the page she/he was asking for in the first place, and not always default. This means that the user can bookmark protected pages on the site, among other things.

**Emailing users from the site:**

To be able to send emails and notifications automatically from the site on a local server, we had to configure the server to allow the relay from the localhost, i.e. we have to add the 'localhost' address (127.0.0.1) to the list of the computers/server that have been granted the permission to relay e-mail through this SMTP virtual server, as in figure 9 below.

This step is needed only once and if we need to run the website on a local machine, otherwise, the application is running and working fine from the project website [www.alooza.com](http://www.alooza.com) as it is already configured.
5.5.2 Main Pages and functions of the Application:

In this section a number of Web pages/functions of the system will be demonstrated including some snapshots from the system working.
Home page (Default.aspx):
This is the entry point of the Website, where it has been made obvious the two major entry points for the main stake holders; the job seekers, and the employers. It is possible to navigate to other resources and pages from this page, see figure 10.

The administrator can also log in from the same log in form in this page, the system can differentiate the type of the user, and hence, to where should this user be redirected, depending on the user type.

After log in, the system creates the suitable session variables, which include the user name, user actual name, type, and ID, those variables used by different pages of the system all the time to make sure that no page can be accessed by any user other than the account owner, they are also used to make sure that pages can be accessed by corresponding type of users where applicable.

Registering new users (RegisterEmployer.aspx and RegisterJS.aspx):
There are 2 separate registration pages for both the job seeker and the employer.

Both types of users start the registration process by filling basic account information and log in details, then the other profile information including education, experience, and contact information for the job seeker, and company information for the employer and so on.
After filling necessary information by the user, the system makes sure that all information entered is correct and valid where it is possible to validate, validation of simple data are done on the client side via client-side script functions (JavaScript) which makes the page more responsive, then the system checks whether the username (email) is taken by another user or not, and so on, after completing the registration process and inserting data in corresponding tables, the user is redirected to her/his account home page depending on the user type.

In the case of the job seeker, the user will be redirected to a page that lists the available relevant vacancies ordered by vacancy rank (VR) mentioned before, the users can choose the vacancy they want, or simply navigate to other pages and functions.

It is a common practice to have only one site administrator, and in addition to the security concerns, i.e. obvious business requirements, there is no registration form for the administrators of course, however, it is needed sometimes to have more than one person having an administrator account, this function (adding administrators) is only available for the administrator after log in.

**Updating users' profiles (UpdateEmployerProfile.aspx, UpdateJobSeekerProfile.aspx, and UpdateAdminProfile.aspx):**

Whether for an administrator, job seeker, or employer, the system enables the users to update their information, login details, contact details, or profile information in any time.

The system shows a form similar to the registration form, but filled with the current account details of the current user, enabling her/him to update only the details she/he needs to update, then, a client-side script is used to validate data as in the case of registration, corresponding tables being updated, suitable session variables being updated, and the user is redirected to the account home page, see figure 11 below.
Adding vacancies (AddVacancy.aspx):

In the case of employer, it is important to add the available (or future) vacancies to the system so to make them available for job seekers to search and browse.

It is possible to make vacancies available within a certain period of time by filling the 'Available from' and 'Available to' fields, this feature enables the employers to enter vacancies that will be available in a known period of time and not necessary from now, this kind of vacancies can not be seen by other job seekers unless within that period intended by the job provider, when the employer leaves these fields blank, the system assumes that the vacancy is available from now to the next year automatically, by having this feature, we guarantee that the old vacancies will not be listed after the 'expiry date' especially if the employer forgot to delete them.

The employer can add many other details about the vacancy, such as experience, education level required, and position details, these details make it easier for the job seeker to get exactly the vacancy suitable for her/him especially when using the search functions included within the system.

After adding the vacancy, the system lists a number of potential job seekers for this vacancy, the system depends on the details just entered, to search for job seekers from the same business category, and/or, have relevant experience, and/or living in the same area, the system shows short information about them, in the same time, it enables the employer to see full...
details about each one of them, see the job seeker CV, print it, or even to
directly contact her/him.

![Image of job posting interface]

Figure 12: Adding new vacancy

**Editing/deleting vacancies (EditVacancy.aspx and DeleteVacancy.aspx):**
These two options are available to the employers when they use the 'Show My Vacancies' options, here the users can see the vacancies belonging to them, it is possible to edit the information of any of these vacancies, or to delete any one of them, bearing in mind that each employer can only edit/delete her/his own vacancies, and even if she/he tries to access another vacancy by changing the parameters in the URL, a warning message appears and the system will not enable her/him to do that.

**Showing and printing CVs (ShowCV.aspx):**
This function is available to all users with different access rights, for example, an employer can browse or search the job seekers by different ways and see their CV's, an employer can also get a direct link by email, (protected by account password) to see the CV of any job seeker who have applied to a vacancy provided by this employer, the system administrator can do similar things, while the job seeker can only see (and edit of course) her/his own CV which is based on her/his profile data saved previously, as the CV is shown, the user can print it directly from the system, see figure 13 below.
The system does not force its own way, format, or template when showing and printing this information, but it also gives the job seeker the ability to upload a document of curriculum vitae and covering letter, and for the employers and site administrators wanting to view or print them, it gives a link for this document, this is a very useful feature for the job seekers especially the ones who do not have Web space to save their CVs.

Note: the above feature was not available from the beginning, until many users asked for it during the evaluation of the system, for more information about this, refer to chapter 6.

**Browsing the vacancies (BrowseVacancies.aspx):**

Administrators and job seekers can browse the available vacancies in the database according to the business category of that vacancy, and then the user can get more details about a particular vacancy and its provider/employer as well as the required contact details.
Figure 14: Browsing vacancies by category

The job seeker can apply/show interest in any vacancy by using the 'Apply to this vacancy' option, which sends this interest directly to the employer who is providing this job.

The employer can see their own vacancies as well.

Applying to a vacancy (ApplyToVacancy.aspx):

The job seeker can see the details of the available vacancies in different parts and pages of the application; whether using browse vacancies categories function, automatic, or even as results of the custom search, in any one of these when the job seeker is interested in a vacancy, it is just one click and the corresponding employer receives an instant email notifying her/him of the new applicant with some details about this job seeker as well as a link to get more information and the full CV of her/him, see figure 15.
Browsing applicants (BrowseJobSeekers.aspx):
As in the 'Browse Vacancies' function for job seeker and administrators, this feature is available for users of the types: employer and administrator, the user can browser the job seekers by category, according to their most recent job (if applicable), then the details of those vacancies are displayed, enabling the user to contact that person, see and/or print the person's CV by simply clicking on her/his name, see figure 16.
**Automatic search (VacanciesAutoSearch.aspx):**

This is one of the most important features in the system, where the job seeker can get a list of relevant vacancies according to her/his own skills and experience history.

At the beginning, the system looks in the database for relevant fields in both of the current job seeker information, and the vacancies information for the vacancies available now, it compares the corresponding fields, and calculate a value (called Vacancy Rank-VR) for each available vacancy according to the equation and technique mentioned in chapter 1, and then the system organizes the required data for these vacancies in a temporary data structure of array type, sorts and reorders the records in this array according to the value of VR calculated before for each one of these records from the higher to the lower, and finally it builds the HTML code necessary to display these details in useful shape and list them to the job seeker.

As shown in figure 17, the system also shows the percentage value of VR to the user under 'Relevancy' title, which also helps the job seeker to realize the order of these vacancies.

*Figure 17: Listing ordered relevant vacancies to the job seeker*
And as in other occasions where the user gets a list of vacancies, the system shows only the important fields to keep the page in a neat layout, allowing the users, if they are interested, to get more details about the vacancy, the vacancy provider (by clicking on the company name), or to apply for any one of them directly from the same page.

**Custom and advanced search (SearchVacancies.aspx):**

Using this page, a job seeker or even the administrator of the site can search the available vacancies by certain keyword and/or category.

After filling the search form by the user, the system looks in the main fields of the vacancies for that keyword, including the position, location, description, preferred educational level, requirements, and so on, and then displays the results that match search terms.

In the case of the job seeker, in addition to the normal search function, the vacancy rank (VR) is calculated for each vacancy that matched the search terms, then reordered from the most relevant to the least, i.e. two stages of search is done here, see figure 18 below for examples of custom search done by a job seeker.

The users can have a custom search done on all of the vacancies fields, or they can specify keywords for some specific fields.
Administrator options and control of the system:

There are some functions available to the site admin, such as editing/deleting business categories, these categories are used to classify vacancies, to classify job seekers according to the recent job, to classify employers according to the industry or business category, and so on.

The other option enables the administrator to edit the adverts shown in the right hand corner of the site, which appears to all users, including visitors of the site, all the time, here the administrators can put HTML code for any
message they want to deliver to those people, they can use it also as advertising space.

Adding new administrators (AddAdmin.aspx):
As we mentioned before, usually, there is only one administrator for the site, created manually at the beginning of the project development, however, it is also useful to enable this administrator to add other administration accounts to be used by other users, in the company that operates the site.
Here, the administrator can fill the username (email), real name, and password of the new administrator, validation is done as usual as in registering new job seeker or employer. The new administrator automatically inherits the same privileges of the first administrator, but cannot add administrators by himself; this keeps the full control of the site with only one person, of course this restriction could be a business requirement, but in real life, this feature (or restriction) could be modified easily if wanted.

Administrator mass email function (MassEmail.aspx):
The site administrators can send mass emails to other users or a group of them.
In this page, administrators can choose one group of users to target or more, these groups are the three user types; administrators, employers, or job seekers, it contains a normal form to fill in the subject of the message and the message, then the message will be delivered to the inboxes of the targeted users they entered upon registration.

Browsing employers (BrowseEmployers.aspx):
Job seekers can browse the available employers in the database by their business category, this feature is also available for administrators which is a useful function for them as it enables them to get information about the site activity in terms of the companies participating, when showing employers in certain category, users can get full details about these employers, or even contact them directly by email.
Search employers (SearchEmployers.aspx):
Job seekers and administrators can search for employers by keyword, and list them, as in the previous function, users can get more details about the listed employers or contact them.

Contact the administrator (Contact.aspx):
This page is available for all, including visitors, where they can send messages, questions, and feedbacks to the site administrator directly. When this form is being filled by a registered user, the real name and email of the user is included automatically.

Other pages:
There are some other simple pages, such as the help page, about page, and privacy policy page.
We have demonstrated the success criteria of the project in chapter 3, and here we will evaluate both the work done in the project, and the approach taken, to indicate whether those goals of the project have been achieved, via a solid evidence and data.

6.1 Testing and evaluation

Usability inspection methods are used by human factors professionals and software developers to inspect, evaluate, and assess the usability of a system, without testing real users.

Inspection methods are typically considered to be less expensive and quicker than other methods because they do not require the cost associated with setting up and conducting usability tests with actual users [40]. These methods are generally based on design guidelines, best practices, cognitive theories, and other theoretical frameworks, and this is what we have done in the first stage of the evaluation.

The first stage of the evaluation was the self evaluation and technical testing which has been carried out without the help of external users, and this part included validating the XHTML code, testing the system with different kinds of inputs, debugging, and other design issues. After that, the system was tested working from different locations, platforms, and browsers. In this stage the system was valid.

Then we used some known methods for interface evaluation, mainly Jakob Nielsen heuristic evaluation, which is a method for structuring the critique of a system using a set of relatively simple and general heuristics [43], and one of the most popular and theoretically well-defined approaches [42], which we are
going to describe below with the reference to the system, and as Nielsen lists them in his Website useit.com.

During the first stage of the design process we have created a prototype of the system and we evaluated it using the ten heuristic rules of J. Nielsen [44]. Since our prototype has a high fidelity prototype and we ran the evaluation cycle, after a lot of work spent on design we did not come across serious violations of the rules but just minor details which we solved after that. Additionally, the nature of the system is such that allows extensive input from the user; based on this fact the system behaves according to the rules in all of the testing experiments that we have run through. Below we present the ten heuristic rules and our findings from the system evaluation against each of these rules.

1. Visibility of the system status
   The system in its present form is executing many processes and functions; the user is providing information that requires process to the system. The response of the system to the user’s actions (clicks etc.) is direct and accurate; the users are always aware of the system’s status since what is presented to them represents the relevant status of the system (in each case), the system always shows the name of the user on the top of the page, and it also shows the links available to the user to the left hand side all the time. Additionally, the functions and sub functions selected are highlighted by a header text in the top of the page contents, as well as the title of the web page itself.

2. Match between system and the real world
   The links of the functions that is presented by the system is broken down into a small number (around 10 or less depending on the type of the user). There is no confusion or overlap of the content; additionally the language that has been used is simple, and is kept to the minimum to allow the main content of the pages to come from the database.
3. User control and freedom
Users can navigate easily within the system and find the information that they want. The main navigation bar (vertical) is always visible.

4. Consistency and standards
All the information of the system is presented to the user in a consistent manner. The design layout is the same through the different pages. There is a standard way of interaction with the system through buttons and links and Web controls and has been kept in all of the system’s pages.

5. Error prevention
Since there is input of information in different places from the users to the system, error prevention was a major concern, which is solved by using validation scripts in forms, and by using drop down lists where applicable to allow the user to chose rather than to type.
On the navigation side, when the user navigates into a non-desirable link; this can be easily corrected because the main navigation bar is always visible.

6. Recognition rather than recall
Usually, this rule is met automatically in Web projects, however, we could use icons instead of text links but we wanted to keep the pages as simple as possible, and we have used close-to-the-main-contents and easy-to-locate links (Fitt’s law [45]).

7. Flexibility and efficiency of use
This rule is met by the fact that this was proved by the feedbacks and notes given in the survey by most of users (see details below), and some of these elements that make the systems use efficient have been mentioned in previous rules.

8. Aesthetic and minimalist design
We have tried throughout the whole design process to keep the system’s layout simple and light, we used suitable and clear font size. In fact most of the users were OK and satisfied with the design according to the survey, very
few of them did not like the colors but this is a personal preference and may vary from one person to another.

9. Help users recognize, diagnose and recover from errors
This rule has been met by using the validation scripts running when users are filling forms, and by giving interactive messages to the users when they try to do something wrong.

The system can be used without documentation; however, we provided a help page for explaining functions, and a feedback page for users' questions if needed.

From the above, we can see that the system design does not violate any of the ten Nielsen heuristics.

There are many other issues to care about, suggested by many Web projects tests and evaluation processes [16][38][22], these were not included in the survey, as they are just reminders of certain elements to be included in the website, most of them are design elements, this kind of issues were tested and applied in the website, and so, there was no need to ask users about them, examples of these reminders include taking care about different screen sizes and resolutions in design, printing, misspellings, technologies used, pages validation, the use of CSS, navigation, cross-platforms and browsers compatibility, forms usage, and so on. All of these things were taken into consideration in the design stage.

The system introduced two main functions; the instant suggestions after registering and adding data, the other is the ranking algorithm for the vacancies, and to prove the success of our system approach in these functions and others, we had to test it by external random real users, and this is what has been done in the second stage of the testing.
In the second stage, a survey was conducted with random users (30+), as mentioned in chapter 3, the following table 3, shows the responses for the 12 multiple choice questions, along with each question, the percentage of satisfied users, i.e. who gave the system 50% positive feedback or more for that question according to our success criteria, in some questions (with *) this criteria was increased to 75% to reflect the meaning of satisfaction for that question.

Alooza Online Recruitment System Evaluation Survey
Response Summary Report
Date Run: Sep 04, 2007 09:00

1. First Impression of the Website:

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>17</td>
<td>57.00%</td>
</tr>
<tr>
<td>Excellent</td>
<td>8</td>
<td>27.00%</td>
</tr>
<tr>
<td>Neutral</td>
<td>4</td>
<td>13.00%</td>
</tr>
<tr>
<td>Fair</td>
<td>1</td>
<td>3.00%</td>
</tr>
<tr>
<td><strong>Total Responses:</strong></td>
<td><strong>30</strong></td>
<td><strong>100.00%</strong></td>
</tr>
<tr>
<td><strong>Number of satisfied users:</strong></td>
<td><strong>29</strong></td>
<td><strong>96.67%</strong></td>
</tr>
</tbody>
</table>

2. The links and sections organizing in website pages is:

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Somehow organized</td>
<td>13</td>
<td>43.00%</td>
</tr>
<tr>
<td>Very well organized</td>
<td>12</td>
<td>40.00%</td>
</tr>
<tr>
<td>Neutral</td>
<td>5</td>
<td>17.00%</td>
</tr>
<tr>
<td><strong>Total Responses:</strong></td>
<td><strong>30</strong></td>
<td><strong>100.00%</strong></td>
</tr>
<tr>
<td><strong>Number of satisfied users:</strong></td>
<td><strong>30</strong></td>
<td><strong>100.00%</strong></td>
</tr>
</tbody>
</table>

3. Functions available for the job seeker:

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Most of what is needed</td>
<td>21</td>
<td>70.00%</td>
</tr>
<tr>
<td>Neutral</td>
<td>4</td>
<td>13.00%</td>
</tr>
<tr>
<td>All of what is needed</td>
<td>3</td>
<td>10.00%</td>
</tr>
<tr>
<td>Fair</td>
<td>2</td>
<td>7.00%</td>
</tr>
<tr>
<td><strong>Total Responses:</strong></td>
<td><strong>30</strong></td>
<td><strong>100.00%</strong></td>
</tr>
<tr>
<td><strong>Number of satisfied users:</strong></td>
<td><strong>28</strong></td>
<td><strong>93.33%</strong></td>
</tr>
</tbody>
</table>

4. Browse Vacancies categories function is:

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>14</td>
<td>47.00%</td>
</tr>
<tr>
<td>Very useful</td>
<td>13</td>
<td>43.00%</td>
</tr>
<tr>
<td>OK</td>
<td>3</td>
<td>10.00%</td>
</tr>
<tr>
<td><strong>Total Responses:</strong></td>
<td><strong>30</strong></td>
<td><strong>100.00%</strong></td>
</tr>
<tr>
<td><strong>Number of satisfied users:</strong></td>
<td><strong>30</strong></td>
<td><strong>100.00%</strong></td>
</tr>
</tbody>
</table>
5. My CV function is:

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>13</td>
<td>43.00%</td>
</tr>
<tr>
<td>Good</td>
<td>9</td>
<td>30.00%</td>
</tr>
<tr>
<td>Very useful</td>
<td>7</td>
<td>23.00%</td>
</tr>
<tr>
<td>Fair</td>
<td>1</td>
<td>3.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30
Number of satisfied users: 29 96.67%

6. How do you evaluate the search vacancies function in general?

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>15</td>
<td>50.00%</td>
</tr>
<tr>
<td>Very useful</td>
<td>8</td>
<td>27.00%</td>
</tr>
<tr>
<td>OK</td>
<td>7</td>
<td>23.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30
Number of satisfied users: 30 100.00%

7. How do you evaluate the results accuracy given by the search vacancies function?

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accurate</td>
<td>12</td>
<td>40.00%</td>
</tr>
<tr>
<td>Neutral</td>
<td>11</td>
<td>37.00%</td>
</tr>
<tr>
<td>Very accurate</td>
<td>4</td>
<td>13.00%</td>
</tr>
<tr>
<td>Not accurate</td>
<td>3</td>
<td>10.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30
Number of satisfied users: 27 90.00%

8. Do you think the search results were ordered in a way or just random?

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ordered according to my needs</td>
<td>15</td>
<td>50.00%</td>
</tr>
<tr>
<td>Ordered but not according to my expectations</td>
<td>11</td>
<td>37.00%</td>
</tr>
<tr>
<td>Not ordered correctly</td>
<td>3</td>
<td>10.00%</td>
</tr>
<tr>
<td>Not ordered</td>
<td>1</td>
<td>3.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30
Number of satisfied users:* 26 86.67%

9. Automatic search function is:

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>13</td>
<td>43.00%</td>
</tr>
<tr>
<td>Very useful</td>
<td>8</td>
<td>27.00%</td>
</tr>
<tr>
<td>OK</td>
<td>8</td>
<td>27.00%</td>
</tr>
<tr>
<td>Fair</td>
<td>1</td>
<td>3.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30
Number of satisfied users: 29 96.67%

10. How do you evaluate automatic search function results?

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>11</td>
<td>37.00%</td>
</tr>
<tr>
<td>Accurate</td>
<td>10</td>
<td>33.00%</td>
</tr>
<tr>
<td>Very accurate</td>
<td>6</td>
<td>20.00%</td>
</tr>
<tr>
<td>Not accurate</td>
<td>3</td>
<td>10.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30
Number of satisfied users: 27 90.00%
11. How do you evaluate the relevancy of vacancies proposed by the system comparing to your expectations?

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>9</td>
<td>30.00%</td>
</tr>
<tr>
<td>Neutral</td>
<td>8</td>
<td>27.00%</td>
</tr>
<tr>
<td>Excellent</td>
<td>7</td>
<td>23.00%</td>
</tr>
<tr>
<td>Fair</td>
<td>6</td>
<td>20.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30 100.00%
Number of satisfied users: 24 80.00%

12. How do you evaluate the easiness of the system?

<table>
<thead>
<tr>
<th>Value</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very easy</td>
<td>18</td>
<td>60.00%</td>
</tr>
<tr>
<td>Easy</td>
<td>7</td>
<td>23.00%</td>
</tr>
<tr>
<td>OK</td>
<td>5</td>
<td>17.00%</td>
</tr>
</tbody>
</table>

Total Responses: 30 100.00%
Number of satisfied users: 30 100.00%

Table 3: Survey users' responses

From the answers above and from the number of satisfied users according to our criteria, we notice that there is an obvious users' satisfaction.

The following table 4 shows a summary of these answers for each user (the answer in each of these questions returned to numeric value as a rating from 1 to 5 to enable calculations), along with the summation of points that the system gained from each user, and the overall average mark of the system.
Table 4: Summary of answers ratings

<table>
<thead>
<tr>
<th></th>
<th>3</th>
<th>4</th>
<th>3</th>
<th>4</th>
<th>2</th>
<th>2</th>
<th>4</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>4</th>
<th>39</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>32</td>
</tr>
<tr>
<td>16</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>41</td>
</tr>
<tr>
<td>18</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>33</td>
</tr>
<tr>
<td>19</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>36</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>28</td>
</tr>
<tr>
<td>21</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>27</td>
</tr>
<tr>
<td>22</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>45</td>
</tr>
<tr>
<td>23</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>35</td>
</tr>
<tr>
<td>25</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>37</td>
</tr>
<tr>
<td>26</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>33</td>
</tr>
<tr>
<td>27</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>37</td>
</tr>
<tr>
<td>28</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>37</td>
</tr>
<tr>
<td>29</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>30</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

Overall average of the system: **35.7**

Again, from table 4, we also notice a user satisfaction, by the fact the system got an average of 35.7 points out of 60, for the targeted users, which is 59.50%, which means that our system got more than 50% of positive feedback, and thus, we can say: the system is successful.

After that, and even after the success of the system was clear, we had to go through the essay questions included within the same survey, in order to know the comments that the users have on the system, the following lists include a summary of these comments.

Notes about the links and sections organizing in website pages:
- Some users stated that the links are clear and easy to find, navigate, and understand, some said that the structure is so normal, but simple and easy for user, some other users stated that the sections were good, clear to see, but visually did not grab them, as we mentioned before, this could be a personal preference and may vary from person to another, however, we could add an option to change the colors used.
in the future, some other users said that the site has a good structure, different pages are coherent, and that it is easy to identifying the pages.

- Some other users suggested to rearrange the order of some links, and to add a nice header and logo to the pages.
- Some users suggested including a feedback section allowing for comment on any difficulty the site has produced, and this function has been built and added for them.

Notes about the relevancy of vacancies proposed by the system comparing to the expectations:

- Some users said that when the database is populated with more vacancies, this will help adding more words to the dictionary, and will lead to more accurate results, and this is simply true!
- Some users stated that the system needs more vacancies in each category but there was a good spread. Again, this resulted from the fact that we had a limited number of vacancies in the evaluation period.
- A user suggested adding a category for 'Social Services/Family Support', and this was added later.
- A user simply said: Almost perfect!

What is the most you liked about the system?

- Some users said that they liked the simplicity, clarity, quickness, easiness to navigate, usefulness, and flexibility of the site and the fact that it contains most of what is needed, and does not take a lot of time to fill the application.
- Some other users liked the vacancies auto search function and described it as 'very helpful'.
- Some users described the system as quick and straightforward, clean, very much usable, well organized, easy to access functions, has simple layout therefore was welcoming and direct.
- Some users liked the back buttons! And some were just happy with the overall concept of a meeting place for employers and job seekers.
Some users thought that the site was excellent.

What is the least you liked about the system?

- Some user did not like the registration form, as it does not provide help tips when users have difficulties in registration.
- Some users did not like the fact that the system does not have many vacancies in the database, but they also said that they think this is not difficult as more and more users use the site and populate the database.
- Some users did not like the simple CV (was before) and suggested to add a CV upload function with cover letter, this function was built later.
- Some few users did not like the auto search function results as it did not seem to function up to their expectations. But it seems that this is due to the limited number of available vacancies, which made the system to list somehow irrelevant results to some users, but of course with low VR value, which gives an impression to the users that these results are not so relevant.
- Some users did not like the fact that the system is only available in English, and suggested to make it a multi language system; this suggestion is legitimate and can be left as future work.
- A user did not like the fact that the system sends the user interest in a vacancy directly to the employer when that user clicks on 'Apply to this Vacancy', and suggested to get more details first. But the reality is, in that page, and in addition to 'Apply to this Vacancy' link, there is additional three links for each vacancy, one gives the user more details about it, the second gives more details about the company that provide this vacancy, and the third, is a link to URL that has more details about the company or the vacancy as listed by the employer.
- A user did not like the colors of the site, and the design, without mentioning details.
What other functions would you like to see in the system? Do you have any suggestion?

Here is a summary of these suggestions:

- Providing a template to prepare the CV.
- Adding a field for the state for people in the United States or Canada. The answer for this is that the system is for all countries, but the user can write the state name in the second line of address, however this not a major issue.
- Having more Job categories, uploaded CV with covering letter. This suggestion was met.
- Adding fields and space for details about previous work, questions, activities, patents, or papers. The answer for this was by adding the Upload CV as a document function, which can include all other details the job seeker wants to mention.
- Improving the visual nature of the site, by having more variation of graphics to grab the attention.
- Some users gave suggestion for business ideas for the site, and these were included in further future work section.

The previous summarized long list of comments was so useful to us, especially the included suggestions, which we tried to apply later in the system; some of these were mentioned in chapter 5 as parts of the system built after the evaluation stage.

The same thing applies on the design side comments, and based on the feedback from the stakeholders, some elements of the page design is suitably modified to meet the stakeholders satisfaction with the page layout, presentation and the navigation structure, all of that after the evaluation stage.

By the numbers mentioned before, and by implementing many of the suggestions given by the users, we can say that the system worked as intended, and so it was successful.
6.2 Reflection

Web page content development needs to take into consideration the stakeholders’ requirements, users’ cognitive abilities, technical issues and considerations, and non-technical issues [41], and this is can only be verified through testing the system by real users, and getting the help and comments on how the problem is solved from them, and that is what we have done in the evaluation stage, which means that the evaluation approach we adopted was a valid one, in addition to the fact that the results of this evaluation process showed that the system was successful, which means that the solution approach used in this system was valid one as well.
Chapter 7

Conclusions & Further Work

In this chapter we will give a brief summary about the dissertation, and highlight the importance of the project as well as what have been achieved. We will also include a discussion of crucial points related to the solved problems in the project and present future work and ideas to improve the system that has been developed.

7.1 Conclusions

As a result of the Web revolution, companies are going global by creating a Web presence, and users are going online, with the huge number of companies and users online, the need for online recruitment systems arise to organize the huge job data available in companies websites and other places, in this project, we built a Web application in the form of profile management portal to help employers and job seekers to find each other in a proper and useful way.

In this report, we introduced some of the popular online job portals, and analysed them critically, by demonstrating the principle of work for each of them and focusing on search and matching functions in these site, and pointing to some limitations they have, such as the human initiative to do the search. Then we introduced our solution to these limitations by building this portal, in the third chapter of this report, we have described the criteria to judge on the success of our system, followed by the design of this project on both data and functions parts, as well as the user interface. After that, we have presented the actual functional requirements, technical implementation, and a number of interesting elements in our system.

In this project, we introduced a new technique of suggesting instant results for both employers and job seekers as they enter their data and requirements,
where the job seekers get a list of suggested vacancies when they register, and for the employers, the system gives a list of potential job seekers when the employer add a new vacancy, this is done automatically after inserting data without the need from user to initiate a certain function.

We also suggested a ranking method depending on the importance of the vacancies to a particular job seeker depending on the importance of each weighted element of the data elements that forms the job seeker profile or the vacancy details, as shown in chapter 1, some of these weights were proposed based on some sources and references, and some were proposed based on logic, it was important to test this equation by getting a feedback from randomly selected users which proved that this technique and the system as a whole was successful.

The two features mentioned above are not available in job portals we tested in chapter 2.

As shown in chapter 6, and from the data collected from those testing users, we noticed that the system got 59.50% of positive feedback, and thus, this is an indication of users' satisfaction, which means that the system introduced ideas that work.

The main feature of our approach is the smart identification of the relevant vacancies, and the quickness of getting results, whether by search functions or email updates.

7.2 Further and future work:

This system was a success in many areas, as shown by the users feedbacks in the evaluation chapter, however there are many improvement and future ideas that can be applied, in this section we will discuss some of them.
On implementation side:
In a real life application, where there is a heavy traffic on the server, it may be recommended to use MS-SQL Server database engine instead of MS-Access, as it works seamlessly with .NET platform, and it is more scalable. We used MS-Access here for demonstrating purposes and for the reason of availability in our development environment.

On structure side:
The system can be rebuilt to be XML managed (semantically valid, and the objects in it are known, so any agent can read from it), with annotated data pages in order to make the system available for other web mining [46] applications.
This requires approaches to provide an integrated interchange of quality metadata that enables agents, and thus the individuals, to access Web content in our system with the most relevant, contemporary data, and knowledge that are both content-rich and practical for decision-making situations.

On search method:
It could be a good idea to proposed to use ontology-based semantic matching [1] capability as part of SQL [9] which facilitates developing ontology-driven database application; rather than using the traditional SQL queries, here is an example of using this technique:

```
SELECT * FROM Vacancies
WHERE ONTRELATED (jobs,
    'IS_A',
    'Part Time Jobs',
    'Jobs_Ontology')=1;
```

Assuming that the ONTRELATED operator here will determine if the two input terms are related by the input relationship type argument by consulting the specified ontology [9] in this example, if they are related, then the operator will return 1, for instance, 0 otherwise, more work has to be done in this area to determine how such kind of ontology can be built or used.
Future ideas on business side (Business model):
As a business application, this portal can be used by a recruitment agency, or a company to get human resources. The portal can be used as a business in its own, the followings are some ways and proposed options the portal can make money with, in real life:

- This model can be supported by adverts, especially the targeted textual ads, relevant to key fields in the displayed data; this can be implemented by having meta-tags of the dynamic generated pages corresponding to the key fields of results data.
- Making data available for free, except the contact details of profiles, jobs, job seekers or employers, and make them available only for premium accounts or for a fee (an idea of applying this is to ask the user to make a premium rate call, or send an SMS, to a number which gives, in return, the access code for these details).
- Selling consultancy based on data in the system.
- Paid and premium listings: Providing paid services such as premium job postings, display ads, target mail, CV database search, branded job search, applicant tracking, banners, and content sponsorships.

The vacancies ranking equation/technique:
In this project, we wanted to demonstrate the idea of listing vacancies (theoretically anything) in the most relevant manner depending on an equation of weighted factors/elements based on their importance in the corresponding field of application (job seeking here), in a more comprehensive real life system, this equation may include more and additional factors/elements that can be located in users' profiles that have information about the job seekers and their previous positions in more details. The same thing applies for the vacancies data on one hand as well as the providers' and employers'.

Many independent elements can be included in this future equation, such as the physical location of the users (the actual location, and not only the data in their profiles), the location of the provider of this vacancy, the number of
available jobs per month for that provider, these details and others can be included in this future equation to give more accuracy for the ranking calculation.

Another interesting thing to implement in the future is the idea of experienced system, i.e. using the system experience in locating suitable job for a job seeker to suggest another job for another job seeker in a similar case, whether by using direct feedback from that first user/users, or by tracing the vacancies that that first user has applied to, and automatically investigating the characteristics of this case, i.e. the characteristics of both the job seeker 1 and the vacancy 1, and save it to apply it (or suggest it) in another similar case (job seeker 2, and vacancy 2), we may need to build a knowledge base for implementing this idea, the similarity between the two cases mentioned here, can be identified by creating two formulas (could be similar to VR), one for the similarity between vacancy 1 and vacancy 2, and the second for the similarity between job seeker 1 and job seeker 2, and this future VR is calculated and saved in kept to be used in a third case and so on, which means an increased accuracy by the time.
References


[22] Galic, Michele; Adams, Jonathan; Bell, Jon A.; Disney, Richard; Kanerva, Ville-Mikko; Matulevich, Steve; Rebman, Kent; Spaas, Philippe (2003), 'Patterns: Applying Pattern Approaches Patterns for e-business Series', IBM RedBooks, IBM Corp.


Appendix A: Project and deliverables proposed plan

Gantt chart for the project phases:

<table>
<thead>
<tr>
<th>ID</th>
<th>Task Name</th>
<th>Start</th>
<th>Finish</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Collect or well document the requirements</td>
<td>03/05/07</td>
<td>14/05/07</td>
<td>1w 5d</td>
</tr>
<tr>
<td>2</td>
<td>Design the system architecture</td>
<td>15/05/07</td>
<td>24/05/07</td>
<td>1w 3d</td>
</tr>
<tr>
<td>3</td>
<td>Design the database</td>
<td>15/05/07</td>
<td>24/05/07</td>
<td>1w 3d</td>
</tr>
<tr>
<td>4</td>
<td>Build the basic portal functions</td>
<td>25/05/07</td>
<td>14/06/07</td>
<td>3w</td>
</tr>
<tr>
<td>5</td>
<td>Implement the search algorithm</td>
<td>15/05/07</td>
<td>26/06/07</td>
<td>1w 5d</td>
</tr>
<tr>
<td>6</td>
<td>Build the advanced portal functions</td>
<td>27/06/07</td>
<td>06/07/07</td>
<td>1w 3d</td>
</tr>
<tr>
<td>7</td>
<td>Initial testing (functionality)</td>
<td>07/07/07</td>
<td>13/07/07</td>
<td>1w</td>
</tr>
<tr>
<td>8</td>
<td>Testing by real users</td>
<td>14/07/07</td>
<td>03/08/07</td>
<td>3w</td>
</tr>
<tr>
<td>9</td>
<td>Implementing feedback ideas</td>
<td>04/08/07</td>
<td>12/08/07</td>
<td>1w 2d</td>
</tr>
<tr>
<td>10</td>
<td>Deployment then delivery, including risk management extra time</td>
<td>13/08/07</td>
<td>07/09/07</td>
<td>3w 5d</td>
</tr>
</tbody>
</table>

Figure 19: Project plan

Note: There was a concurrent writing of the dissertation report for the corresponding parts of the project.
Appendix B: How to run the system?

To run the website there are two options:

⇒ **Online:**

This is the easier way as the website is fully running on this address [http://www.alooza.com](http://www.alooza.com)

All settings and required software are already there, the server runs Microsoft .NET Framework 2.0, under the Web server IIS6.

To test it you have to register as an employer or a job seeker, otherwise, you can use the following ready-to-use log in information, which has been created for testing purpose:

Job Seeker:
Username: j@e4t.net
Password: jpass

Employer:
Username: e@e4t.net
Password: epass

Site administrator:
Username: admin@e4t.net
Password: adminpass

⇒ **Using the local server:**

To run the website on your local machine, you need an operating system such as Windows running IIS6 with Microsoft .NET Framework 2.0, if it is not available on your local machine you can download it from the Microsoft ASP.NET Official site at: [http://asp.net/downloads/essential](http://asp.net/downloads/essential)

You can also download Microsoft Visual Web Developer from the same page if you want to edit or have a look on the code.

After that, you have to copy the website files from the attached CD website folder to the server in this folder `c:\Inetpub\wwwroot\{Alooza-Directory}`

---

*The CD is attached on the back cover of the report.*
The name between brackets is just a suggestion.
Make sure the Web server (IIS) is running.
Navigate from your browser to:
And you must see the home page now.
You can use the log in information mentioned before or register as mentioned above.
Note: the names of folders and/or drives may vary, you may need to change the name of the server or add a port number depending on the setting of your computer.
You may need to configure the server to enable sending emails as described in chapter 5.