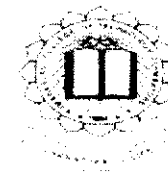


EMPLOYEE RECRUITMENT SYSTEM

By

RAHUL . V

Reg. No. 71203621038

Of

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

JUNE, 2006

Kumaraguru College of Technology

Coimbatore – 641006

Department Of Computer Applications

Bonafide Certificate

Certified that this project report titled **EMPLOYEE RECRUITMENT SYSTEM** is the bonafide work of Mr. **RAHUL.V,** who carried out the research under our supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

N.22
Project Guide

Head of Department

Submitted for the University Examination held on 29-06-2006

29/6
Internal Examiner

29/6/06
External Examiner

May 24, 2006.

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Rahul V is pursuing an internship with Accenture Services Pvt. Ltd.; the period of internship is from January 19, 2006 to July 19, 2006.

During this period, the intern has been working on a project entitled as **"Employee Recruitment System ".**

Till date, we found the intern's conduct satisfactory.

Yours Sincerely

Muthukumar Suruli
India HR Services Centre,
Accenture Services Pvt. Ltd.

ABSTRACT

Maintaining data in a recruitment process of an employee is a long and tedious job. Mapping the recruitment process to a resource demand in a project makes it further tough. Employee Recruitment System is an automation system aimed at automating the demand supply management, starting from creating a demand for a resource to satisfying the demand through recruitment process.

The system starts by recording a demand for a resource for a project. The demand should explain the details such as skill, the operating group, number of resources needed etc. Then individual requests are generated for the resource. The system proceeds to recruitment process, taking a candidate through various tests and interviews. Once the candidate satisfies the needs aptly, he/she would be made a resource of the organization. This meets the particular demand raised earlier.

Employee recruitment system is web-based. The system would be deployed in Windows 2003 Server. .Net is used as a front end and SQL Server 2000 SP3 as the backend. The system will make the process of recruitment much simpler with defined roles for the recruiter, the project manager and the administrator.

ACKNOWLEDGEMENT

I express my grateful thanks to our principal, **Dr. Joseph.V.Thanikal**, and former principal, **Dr. K.K.Padmanabhan**, Kumaraguru College of Technology, Coimbatore, for extending their help by providing all the facilities within our college.

I would like to express my sincere thanks to **Dr.M.Gururajan**, HOD, Department of Computer Applications, Kumaraguru College of Technology, for having permitted me to undertake this project.

I convey my earnest thanks to **Mr. N. Jayakanthan**, Lecturer, Department of Computer Applications, Kumaraguru College of Technology, for his invaluable guidance, support and suggestions throughout the course of this project work.

I also thank **Mr. Nagarajan Dhandapani**, System Analyst, Accenture, for his guidance and support at Accenture Services Pvt Limited, Chennai.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ERS	Employee Recruitment System
RRD	Resource Requirement Details
SR	Sourcing Requisitions
SRS	System Requirements Specification

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DEFINITION

The Recruitment System is a broad process. A lot of information is to be maintained in the process, from requirement of the resource to taking the candidate through the recruitment process, until making the resource on board the company. The current system uses Excel sheets for maintaining the whole data and the recruiter has to manually enter all the details as and when it happens. Moreover a data may need to be changed in multiple locations to reflect a single change. The recruiter has to keep track of all these locations and do the change manually.

Hence the problem wraps up to designing an automated system which will be capable of maintaining the data in an efficient way and help the recruiter and other users enter the details easily. Moreover the system should reflect changes made in one location to all the places corresponding to the same data. The users must also be able to view the data in a comfortable way.

1.2 INTRODUCTION TO ERS

ERS aims at automating the process of recruitment by making the demand supply management easier for a project. The system will first record the details about the project for which the recruitment is to be done. Then the

demand for the project is raised by the project manager. Individual requests are generated based on the number of resources needed, raised by the project manager. These requests are to be satisfied by the recruitment process. The HR of the company will now start the recruitment process by selecting a candidate, conducting tests and interviews on the candidate and finally selecting the candidate if apt for the requirement raised. Each stage of the recruitment process is recorded clearly in the ERS. Once a candidate is selected, he/she is tagged against a particular request and hence the candidate becomes the resource of the company. The resource is then made onboard. This is the scope of ERS.

1.3 ERS -- OVERVIEW

The system goes through a series of steps in handling the recruitment process. The modules generally are,

- Project

  The module records detail about the project for which the recruitment is done. The details include name of the client, the start and end dates etc. The module also handles searching and editing the project.

- Resource Requirement Details

  This module raises a request for the resources. The total number of resources needed, the primary skill, level etc are recorded in RRD. The module also handles the search and edit of request raised.

• Sourcing Requisition

This module creates individual requests for the request raised in the previous step. Individual requests are generated based on the priority of the issue. The module also handles editing and searching of the requests generated.

• Candidate

The role of an HR starts from here. The module handles all processes from adding a new candidate to interviewing, testing etc of the candidate to finally offer the candidate a chance to join the company. The module can also search and edit these details from this module.

• Joiner/ Resource processing

This module stores the joining details of the candidate. A selected candidate is made on board as a resource in this module. The module also allows searching for resources and editing their details.

• Reports

Three basic reports are generated in this system, namely, Demand Sheet, Pipeline report and Position Comparison Report. These reports show the complete details in a grid format.

• Security

This module handles all security issues related to a user. The module handles user creation, role creation etc. Administrator makes use of this module.

CHAPTER 2

SYSTEM ANALYSIS

2.1 WHAT IS SYSTEM ANALYSIS?

System Analysis deals with description of the current system and where problems or opportunities are with a general recommendation on how to fix, enhance, or replace chosen system; explanation of alternative system and justification for the chosen alternative. It deals with studying the system and proposing changes that makes a positive impact to the organization.

A brief requirement is given as input to the analysis process. The requirements are studied and analyzed whether the proposed system is feasible within the current constraints. The result of the analysis is a defined and agreed software requirements specification. This srs further stands as a point of definition which can be referred to any time to know the exact requirements of the system.

2.2 HARDWARE REQUIREMENTS

Development Environment :

512MB RAM

Pentium 4, 2.8 G Hz

2.3 SOFTWARE REQUIREMENTS

User Machine :

Windows XP Professional

Microsoft Internet Explorer 6.0

Web Server Machine :

Windows 2003 Server Enterprise Edition

IIS 6.0

DOT NET Framework 1.1

Microsoft Excel

SQL Server 2000 Client

Database Server Machine :

Windows 2003 Server Enterprise Edition

SQL Server 2000 SP3

2.4 COMMUNICATIONS INTERFACE

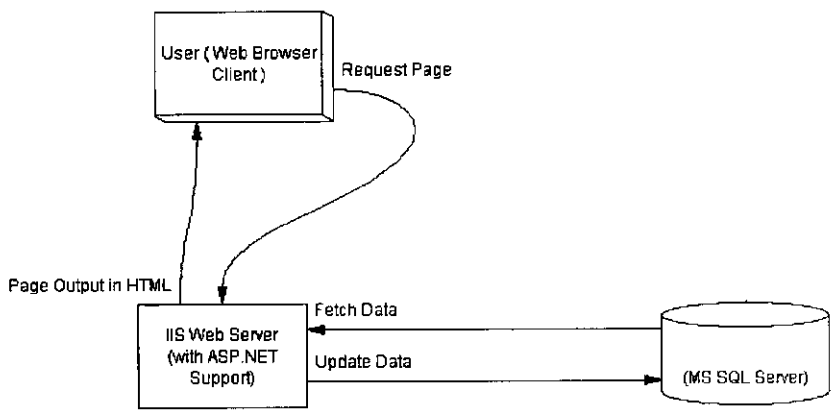


Figure 2.1 Communication Interface

2.5 REQUIREMENTS DEFINITION

2.5.1 Project :

- The system shall have provision for adding, searching, modifying the details of a project.
- The system must have provision to export the results of a search to excel sheet.

2.5.2 Resource Requirement details:

- The system shall have provision for adding, searching, modifying the contents of an existing requirement detail
- The system must have provision to export the results of a search to excel sheet.
- The project related fields should automatically be populated for requirement details.

2.5.3 Sourcing Requisition :

- The system shall have provision for adding, searching, modifying the contents of an existing SR.
- The system must have provision to export the results of a search to excel sheet.
- The project related fields should automatically be populated for SR
- The system shall have provision for generating SRs

2.5.4 Candidate:

- The system shall have provision for adding, searching, modifying the contents of an existing Candidate
- The system must have provision to export the results of a search to excel sheet.
- The system shall give provision for uploading candidate resumes.
- Ability to Add/ Modify/ Delete/ Export the following details of the candidate : -

  Education details
  Skill details
  Interview details
  Offer details
  Document details
  SR details
  Organization history

- Ability to resolve Potential Duplicates to True Duplicates or Original
- True Duplicate Candidates cannot be resolved into Potential Duplicates or Original. If a true duplicate candidate needs to be made original then the only way is to add the candidate as a new candidate
- Ability to tag/ detag a candidate from a candidate request.

- An offer can be made to a candidate only if the candidate attends atleast one face to face interview.

2.5.5 Joiner / Resource Processing :

- The system shall have the ability to search / update joiner information.
- A candidate becomes a joiner only after accepting the offer.
- The system shall have the ability to mark resource onboard/ update resource information.
- Candidate must become a resource only after the candidate is in offer accepted status.
- The system must have provision to export the results of a search to excel sheet.
- Date of Joining is mandatory for making a resource onboard
- Ability to record the renege reason of candidate

2.5.7 Reports : Demand Sheet

- Criteria for demand sheet report are : From date , To date
- Should specify the status of the demanded resource, and other details of the demand.

Reports : Pipeline Report

- Criteria for pipeline report are : From date , To date
- Should specify the details and status of the candidates based on service line.

Reports : Position Comparison Tracker Report

- Based on Organization , graduation and status of the candidate
- Details of the candidate according to the criteria should be shown.

2.5.8 Security :

- Ability to add / edit users
- Ability to add / edit /delete roles

2.6 PROCESS FLOW

The flow of control through various processes identified from the requirements is shown in figure. The requirements are analyzed to decide on the processes in the system and how the control flows through these processes.

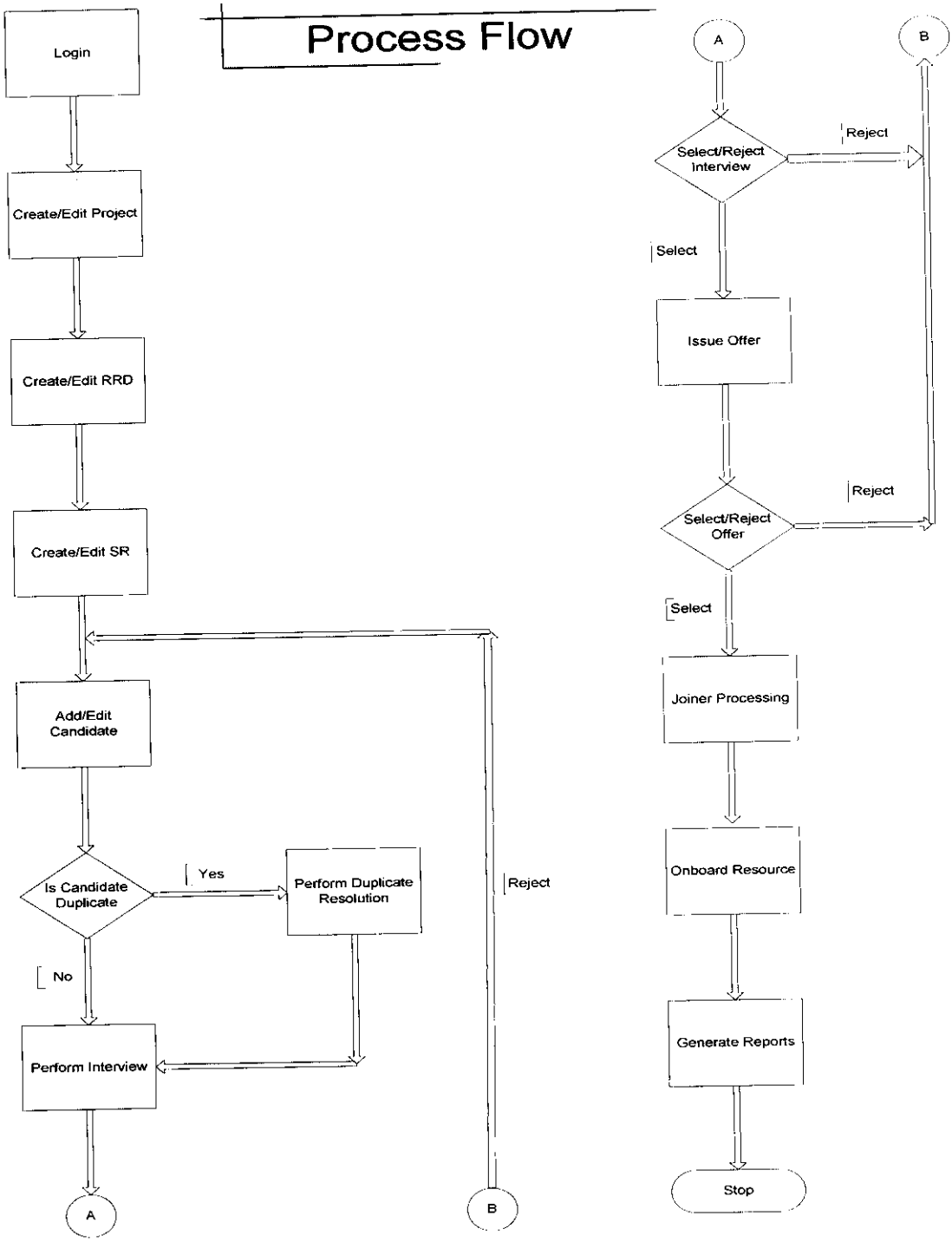


Figure 2.2 Process Flow

CHAPTER 3

SYSTEM DESIGN

3.1 SYSTEM DESIGN OVERVIEW

System Design is the process of transition from analysis to design specifications. System design includes activities like determining detailed requirements, design of data/information flow, design of database, design of user interface, physical design, and design of hardware/software configuration. The input, output, database interactions, files, all should be understood well in the system design process.

3.2 DATA FLOW DIAGRAMS

The flow of data through the system is represented using a data flow diagram. The data flow diagram for ERS has been decided upto level 2.

The Context level diagram shows the overall system with the users who will be interacting with it, being recruiter, project manager and administrator in this case. The project manager has the role of creating a new project, and raising new requests for resources for the created project. The Recruiter does the whole recruitment process. The administrator takes care of the user/ role creation and edits.

Context Level Diagram

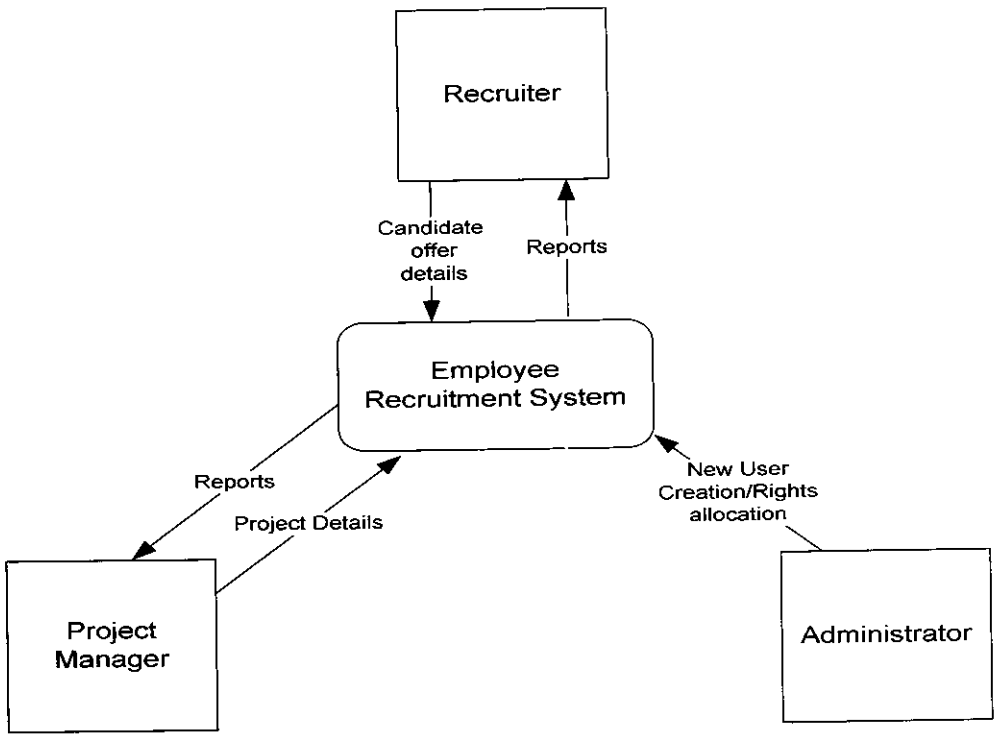


Figure 3.1 Context Level Diagram

The level 1 DFD will explain the major modules in the whole system, i.e., how the data flow between each of these modules. The flow from once a user logins to raising a request, recruiting a candidate to report generation is shown in level 1 of the data flow diagram. The interaction of each process with the corresponding tables is also shown.

Level 1 DFD

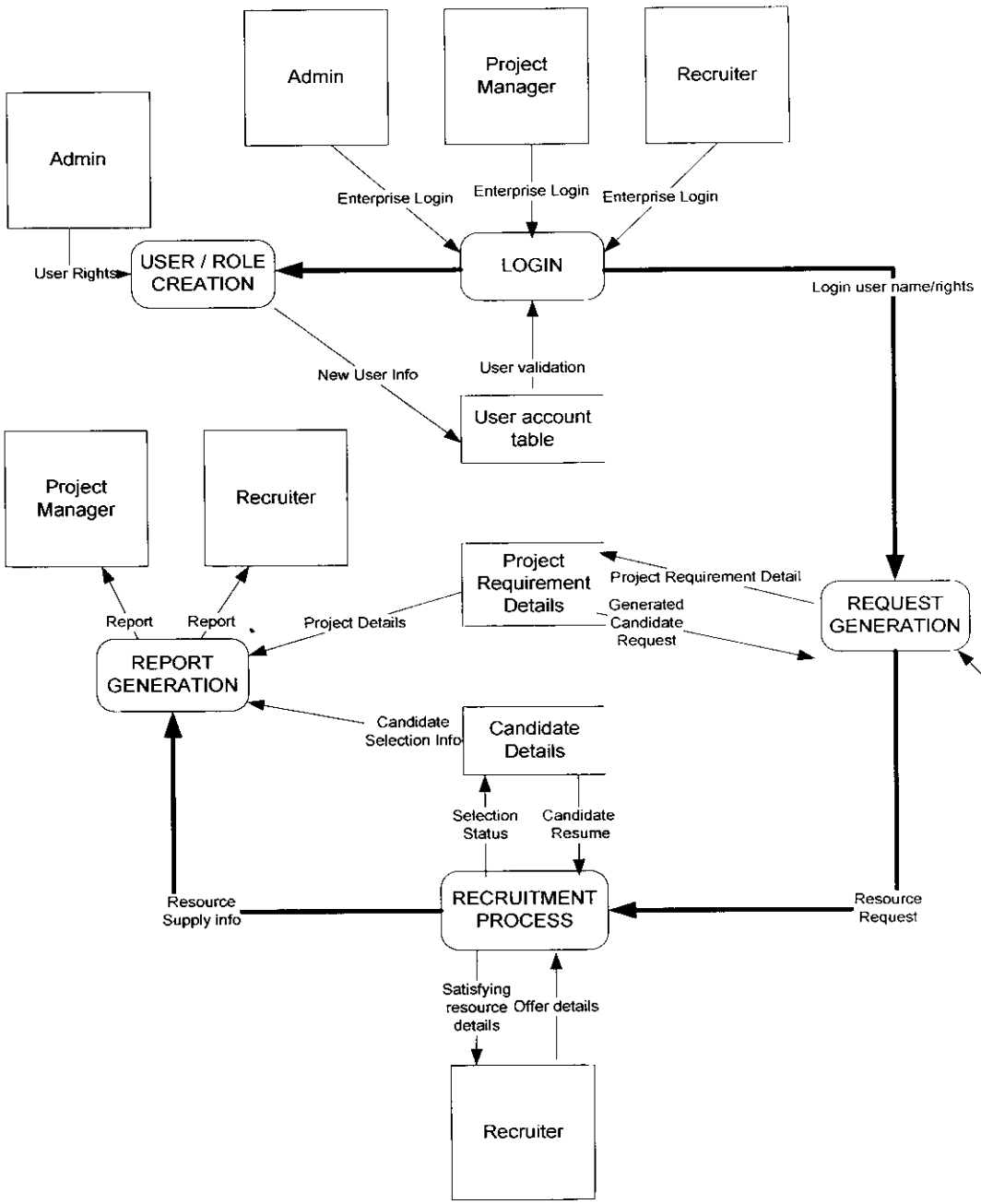


Figure 3.2 Level 1 DFD

The level 2 of data flow diagram shows the detailed processing in these modules. Two modules are expanded to level two, Request Generation and the Recruitment process.

The Request Generation process includes adding a project, creating a Resource Requirement Detail and creating individual Sourcing Requisitions.

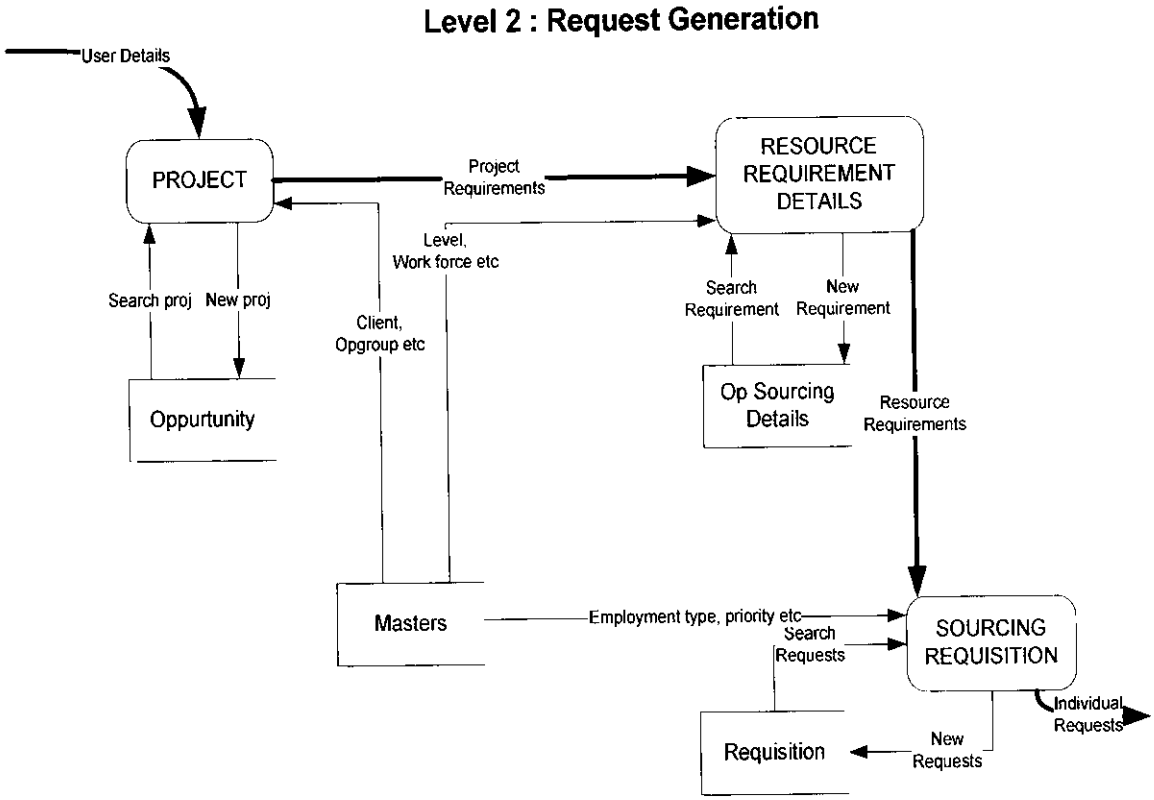


Figure 3.3 Level 2 DFD Request Generation

Similarly, the recruitment process further is broken down to adding a new candidate, interview process, joiner details, resource processing and recording the expenses.

Level 2 – Recruitment Process

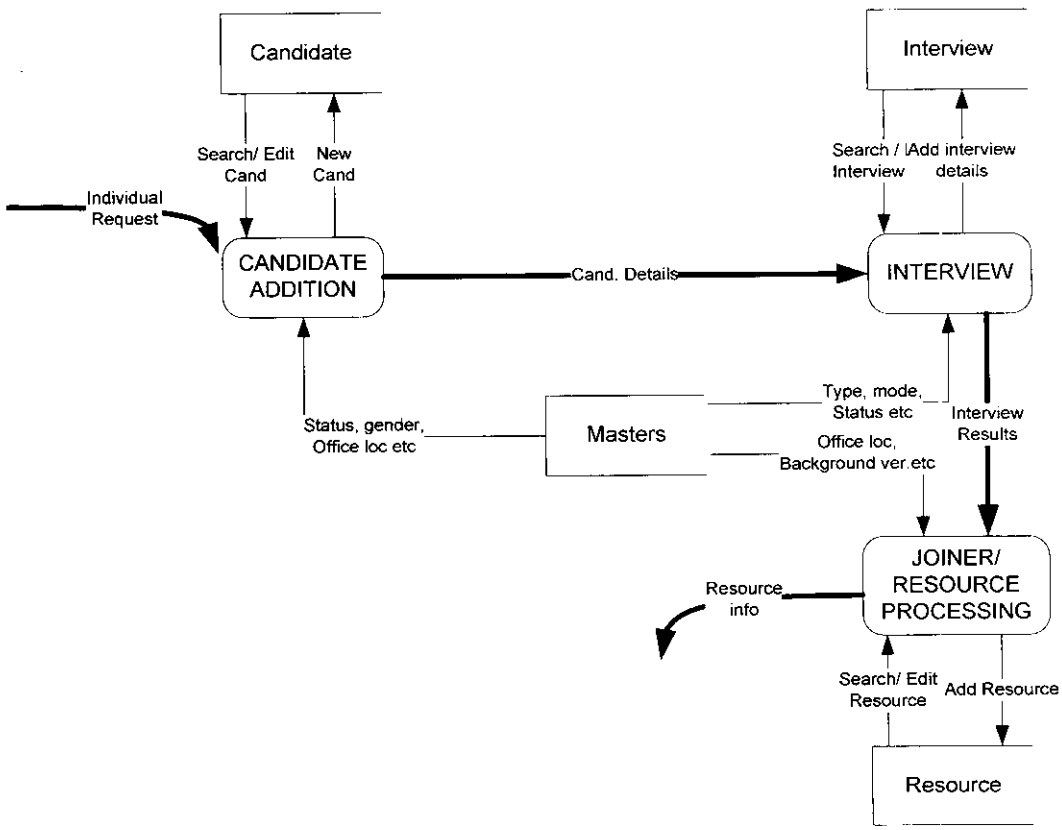


Figure 3.4 Level 2 DFD Recruitment Process

3.3 TABLE DESIGN

Table 3.1 Opportunity (Project Details)

Field Name	Data Type	Key Reference	Not Null
Op_id	Bigint(8)	Primary key	Y
Op_opm_id	Bigint(8)	(op probability master)	Y
op_name	Varchar(100)		Y
op_point_of_contact	Varchar(100)		
Op_project_manager	Varchar(100)		
Op_project_startdate	Datetime(8)		
Op_project_enddate	Datetime(8)		
Op_summary	Varchar(1000)		
Op_cl_id	Bigint(8)	(Client master)	Y
Op_ct_id	Bigint(8)	(Country master)	
Op_og_id	Bigint(8)	(operating group master)	Y
Op_apm_id	Bigint(8)	(approving partner master)	Y
Op_otm_id	Bigint(8)	(opportunity type master)	
Op_slm_id	Bigint(8)	(service line master)	
Op_created_by	Varchar(100)		
Op_created_on	Datetime(8)		
Op_activeyn	Char(1)		Y

Table 3.2 Resource Requirement Detail (Requirement for resource)

Field Name	Data Type	Key Reference	Not Null
Osd_id	Bigint(8)	Primary key	Y
Osd_recvd_date	Datetime(8)		Y
Osd_onboarding_date	Datetime(8)		Y
Osd_enddate	Datetime(8)		
Osd_job_description_filename	Varchar(500)		
Osd_comment	Varchar(500)		
Osd_skill_comments	Varchar(500)		
Osd_res_req_count	Int(4)		Y
Osd_level_comments	Varchar(500)		
Osd_client_approval_reqd	Char(1)		
Osd_work_location	Varchar(100)		
Osd_ol_id	Bigint(8)	(office location master)	Y
Osd_rl_id	Bigint(8)	(resource level master)	Y
Osd_wf_id	Bigint(8)	(workforce master)	Y
Osd_os_id	Bigint(8)	(opportunity status master)	Y

Table 3.2 Continued

Osd_skm_id	Bigint(8)	(skill master)	Y
Osd_ip_id	Bigint(8)	(interviewer pool)	
Osd_activeyn	Char(1)		Y
Osd_created_by	Varchar(100)		Y
Osd_created_on	Datetime(8)		Y
Osd_op_id	Bigint(8)		Y

Table 3.3 Sourcing Requisition (Individual requests)

Field Name	Data Type	Key Reference	Not Null
Rq_id	Bigint(8)	Primary key	Y
Rq_tent_dt_fulfil	Datetime(8)		
Rq_tent_dt_closure	Datetime(8)		
Rq_act_dt_fulfil	Datetime(8)		
Rq_act_dt_closure	Datetime(8)		
Rq_sr_creation_dt	Datetime(8)		
Rq_rcomments	Varchar(500)		
Rq_withdrawn_dt	Datetime(8)		
Rq_grp_no	Bigint(8)		
Rq_et_id	Bigint(8)	(employment type master)	Y
Rq_py_id	Bigint(8)	(priority master)	Y
Rq_osd_id	Bigint(8)	(opsourcing detail)	Y
Rq_au_id	Bigint(8)	(app user)	
Rq_rsm_id	Bigint(8)	(requisition status master)	Y
Rq_swr_id	Bigint(8)	(sr withdrawal reason master)	
Rq_created_by	Varchar(100)		Y
Rq_created_on	Datetime(8)		Y
Rq_activeyn	Char(1)		Y

Table 3.4 Candidate (Basic Candidate Information)

Field Name	Data Type	Key Reference	Not Nulls
Cn_id	Bigint(8)	Primary key	Y
Cn_fname	Varchar(100)		Y
Cn_lname	Varchar(100)		Y
Cn_mname	Varchar(100)		
Cn_dob	Datetime(8)		
Cn_perm_addr	Varchar(500)		
cn_curr_addr	Varchar(500)		
Cn_email	Varchar(100)		Y
Cn_alt_email	Varchar(100)		
Cn_mobile	Varchar(100)		
Cn_resume_doc_name	Varchar(500)		Y
Cn_comments	Varchar(500)		
Cn_prev_acc_emp	Char(1)		
Cn_recvd_dt	Datetime(8)		Y
Cn_ref_name	Varchar(100)		
Cn_ref_email	Varchar(100)		
Cn_curr_location	Varchar(100)		
Cn_other_contact_no	Varchar(100)		
Cn_relevant_exp	Int(4)		
Cn_ref_id	Bigint(8)		
Cn_curr_rstm_id	Bigint(8)	(state_master)	
Cn_msm_id	Bigint(8)	(marital status)	
Cn_csm_id	Bigint(8)	(cnd status master)	
Cn_ct_id	Bigint(8)	(country master)	
Cn_gem_id	Bigint(8)	(gender master)	
Cn_ol_id	Bigint(8)	(office loc master)	
cn_cm_id	Bigint(8)	(city master)	
Cn_au_id	Bigint(8)	(app user)	
Cn_ch_id	Bigint(8)	(channel master)	
Cn_nm_id	Bigint(8)	(nationality master)	
Cn_cdt_id	Bigint(8)	(cnd_dup_types_master)	
Cn_te_id	Bigint(8)	(tot_exp_master)	
Cn_slm_id	Bigint(8)	(service line master)	
Cn_rl_id	Bigint(8)	(resource level master)	
Cn_created_by	Varchar(100)		Y
Cn_created_on	Datetime(8)		Y
Cn_activeyn	Char(1)		Y

Table 3.5 Candidate Education (Education Information)

Field Name	Data Type	Key Reference	Not Null
Ce_id	Bigint(8)	Primary key	Y
Cd_is_max	Char(1)		
Ce_passing_yr	Datetime(8)		
Ce_univ_name	Varchar(100)		
Ce_comments	Varchar(500)		
Ce_gm_id	Bigint(8)	(graduation master)	Y
Ce_spm_id	Bigint(8)	(specialization master)	Y
Ce_cn_id	Bigint(8)	(candidate)	Y
Ce_dm_id	Bigint(8)	(degree_master)	Y
Ce_created_by	Varchar(100)		Y
Ce_created_on	Datetime(8)		Y
Ce_activeyn	Char(1)		Y

Table 3.6 Candidate Skill (Skill Information)

Field Name	Data Type	Key Reference	Not Null
Csk_id	Bigint(8)	Primary key	Y
Csk_comments	Varchar(500)		
Csk_isprimary	Char(1)		
Csk_cn_id	Bigint(8)	(candidate)	Y
Csk_skm_id	Bigint(8)	(skill master)	Y
Csk_created_by	Varchar(100)		Y
Csk_created_on	Datetime(8)		Y
Csk_active_yn	Char(1)		Y

Table 3.7 Candidate Organization (Organization Information)

Field Name	Data Type	Key Reference	Not Null
Co_id	Bigint(8)	Primary key	Y
Co_role	Varchar(100)		Y
Co_compensation	Varchar(40)		Y
Co_worked_from	Datetime(8)		Y
Co_worked_to	Datetime(8)		
Co_immediate_org_yn	Char(1)		Y
Co_cn_id	Bigint(8)	(candidate)	Y

Table 3.7 Continued

Co_cm_id	Bigint(8)	(currency master)	Y
Co_om_id	Bigint(8)	(organization master)	Y
Co_created_by	Varchar(100)		Y
Co_created_on	Datetime(8)		Y
Co_active_yn	Char(1)		Y

Table 3.8 Interview (Candidate Interview Information)

Field Name	Data Type	Key Reference	Not Null
Iv_id	Bigint(8)	Primary key	Y
Iv_interview_dt	Datetime(8)		Y
Iv_comments	Varchar(500)		
Iv_project_spec_yn	Char(1)		
Iv_interview_address	Varchar(500)		
Iv_interviewlocation	Varchar(500)		
Iv_itm_id	Bigint(8)	(interview type master)	Y
Iv_cn_id	Bigint(8)	(candidate)	Y
Iv_imm_id	Bigint(8)	(interview mode master)	
Iv_istu_id	Bigint(8)	(interview status master)	Y
Iv_ip_id	Bigint(8)	(interview pool)	Y
Iv_ilm_id	Bigint(8)	(levels master)	
Iv_created_by	Varchar(100)		Y
Iv_created_on	Datetime(8)		Y
Iv_activeyn	Char(1)		Y
Iv_op_id	Bigint(8)	(oppurtunity)	

Table 3.9 Interviewer Pool (Interviewer Information)

Field Name	Data Type	Key Reference	Not Null
Ip_id	Bigint(8)	Primary key	Y
Ip_comments	Varchar(500)		
Ip_interviewer_name	Varchar(100)		Y
Ip_interviewer_email	Varchar(100)		
Ip_avail_from_dt	Datetime(8)		
Ip_avail_to_dt	Datetime(8)		
Ip_itm_id	Bigint(8)	(interview type master)	Y
Ip_created_by	Varchar(100)		Y
Ip_created_on	Datetime(8)		Y
Ip_active_yn	Char(1)		Y

Table 3.10 Offer Details (Offer Information)

Field Name	Data Type	Key Reference	Not Null
Od_id	Bigint(8)	Primary key	Y
Od_offer_made_dt	Datetime(8)		
Od_fixed_compensation	Varchar(40)		
Od_fixed_comp_currency	Bigint(8)		
Od_total_compensation	Varchar(40)		
Od_total_comp_currency	Bigint(8)		
Od_notice_period_buyout	Varchar(40)		
Od_notice_period_buyout_currency	Bigint(8)		
Od_joining_bonus	Varchar(40)		
Od_joining_bonus_currency	Bigint(8)		
Od_relocation_exp	Varchar(40)		
Od_relocation_exp_currency	Bigint(8)		
Od_hot_skill_bonus	Varchar(40)		
Od_hot_skill_bonus_currency	Bigint(8)		
Od_move_to_next_level_on	Varchar(40)		
Od_offer_accept_date	Datetime(8)		
Od_actual_joining_date	Datetime(8)		
Od_offer_declined_date	Datetime(8)		
Od_comments	Varchar(500)		
Od_var_compensation	Varchar(40)		
Od_var_comp_currency	Bigint(8)		
Od_notice_prd_buyout_proj_paid_yn	Char(1)		
Od_join_bonus_proj_paid_yn	Char(1)		
Od_relocation_required_yn	Char(1)		
Od_hotskill_bonus_proj_paid_yn	Char(1)		
Od_tot_exp_positioning	Int(4)		
Od_tentative_joining_date	Datetime(8)		
Od_relocation_from	Varchar(100)		
Od_joining_loc Od_rq_id	Varchar(100) Bigint(8)	(requisition)	
Od_cn_id	Bigint(8)	(candidate)	Y
Od_odrm_id	Bigint(8)	(offer decline reason master)	
Od_et_id	Bigint(8)	(employment type master)	
Od_ilm_id	Bigint(8)	(levels master)	
Od_au_id	Bigint(8)	(app user)	
Od_created_by	Varchar(100)		Y
Od_created_on	Datetime(8)		Y
Od_activeyn	Char(1)		Y

Table 3.11 Candidate documents (Documents Information)

Field Name	Data Type	Key Reference	Not Null
Cd_id	Bigint(8)	Primary key	Y
Cd_submitted_on	Datetime(8)		Y
Cd_comments	Varchar(500)		
Cd_doc_path	Varchar(500)		Y
Cd_dtm_id	Bigint(8)	(doc_type_master)	Y
Cd_cn_id	Bigint(8)	(candidate)	Y
Cd_created_by	Varchar(100)		Y
Cd_created_on	Datetime(8)		Y
Cd_active_yn	Char(1)		Y

Table 3.12 Candidate SR mapping (Mapping candidate to particular SR)

Field Name	Data Type	Key Reference	Not Null
Csm_id	Bigint(8)	Primary key	Y
Csm_rq_id	Bigint(8)	(requisition)	Y
Csm_cn_id	Bigint(8)	(candidate)	Y
Csm_created_by	Varchar(100)		Y
Csm_created_on	Datetime(8)		Y
Csm_active_yn	Char(1)		Y

Table 3.13 Candidate duplicate (Duplicate record information)

Field Name	Data Type	Key Reference	Not Null
Cnd_id	Bigint(8)	Primary key	Y
Cnd_fname	Varchar(100)		Y
Cnd_lname	Varchar(100)		Y
Cnd_mname	Varchar(100)		
Cnd_dob	Datetime(8)		
Cnd_perm_addr	Varchar(500)		
Cnd_curr_addr	Varchar(500)		
Cnd_email	Varchar(100)		Y
Cnd_alt_email	Varchar(100)		
Cnd_mobile	Varchar(100)		
Cnd_resume_doc_name	Varchar(500)		Y
Cnd_comments	Varchar(500)		
Cnd_prev_acc_emp	Char(1)		
Cnd_recvd_dt	Datetime(8)		Y

Table 3.13 Continued

Cnd_ref_name	Varchar(100)		
Cnd_ref_email	Varchar(100)		
Cnd_curr_location	Varchar(100)		
Cnd_other_contact_no	Varchar(100)		
Cnd_relevant_exp	Int(4)		
Cnd_ref_id	Bigint(8)		
Cnd_curr_rstm_id	Bigint(8)	(state master)	
Cnd_msm_id	Bigint(8)	(marital status)	
Cnd_csm_id	Bigint(8)	(cnd status master)	
Cnd_ct_id	Bigint(8)	(country master)	
Cnd_gem_id	Bigint(8)	(gender master)	
Cnd_ol_id	Bigint(8)	(office loc master)	
Cnd_cm_id	Bigint(8)	(city master)	
Cnd_au_id	Bigint(8)	(app user)	
Cnd_ch_id	Bigint(8)	(channel master)	
Cnd_nm_id	Bigint(8)	(nationality master)	
Cnd_cdt_id	Bigint(8)	(cnd_dup_types_master)	
Cnd_te_id	Bigint(8)	(tot_exp_master)	
Cnd_slm_id	Bigint(8)	(service line master)	
Cnd_rl_id	Bigint(8)	(resource level master)	
Cnd_created_by	Varchar(100)		Y
Cnd_created_on	Datetime(8)		Y
Cnd_cn_id	Bigint(8)	(candidate)	Y
Cnd_activeyn	Char(1)		Y

Table 3.14 Candidate duplicate relation (Potential Duplicate details)

Field Name	Data Type	Key Reference	Not Null
Cdr_id	Bigint(8)	Primary key	Y
Cdr_parent_id	Bigint(8)		Y
Cdr_child_id	Bigint(8)		Y
Cdr_created_by	Varchar(100)		Y
Cdr_created_on	Datetime(8)		Y
Cdr_cn_id	Bigint(8)	(candidate)	Y

Table 3.15 Resource (Resource Information)

Field Name	Data Type	Key Reference	Not Null
Rs_id	Bigint(8)	Primary key	Y
Rs_fname	Varchar(100)		Y
Rs_mname	Varchar(100)		
Rs_lname	Varchar(100)		Y
Rs_acc_email_id	Varchar(100)		
Rs_personnel_no	Varchar(18)		
Rs_end_date	Datetime(8)		
Rs_resign_date	Datetime(8)		
Rs_work_phone	Varchar(100)		
Rs_emergency_contact	Varchar(100)		
Rs_emergency_no	Varchar(100)		
Rs_rec_remarks	Varchar(100)		
Rs_emergency_cnt_relation	Varchar(100)		
Rs_transit_accom_required_yn	Char(1)		
Rs_careercounsellor	Varchar(100)		
Rs_onboard_remarks	Varchar(100)		
Rs_rq_id	Bigint(8)	(requisition)	
Rs_ol_id	Bigint(8)	(office loc master)	Y
Rs_cn_id	Bigint(8)	(candidate)	Y
Rs_od_id	Bigint(8)	(offer_details)	Y
Rs_bgv_id	Bigint(8)	(background verification master)	
Rs_rrr_id	Bigint(8)	(resource_renege reason master)	
Rs_created_by	Varchar(100)		Y
Rs_created_on	Datetime(8)		Y
Rs_active_yn	Char(1)		Y

Table 3.16 Applinks (Available links information)

Field Name	Data Type	Key Reference	Not Null
Al_id	Bigint(8)	Primary key	Y
Al_desc	Varchar(500)		
Al_link_name	Varchar(100)		Y
Al_url	Varchar(500)		
Al_grp_no	Int(4)		
Al_grp_name	Varchar(100)		
Al_created_by	Bigint(8)		Y
Al_created_on	Datetime(8)		Y
Al_active_yn	Char(1)		Y

Table 3.17 Appuser (Users Information)

Field Name	Data Type	Key Reference	Not Null
Au_id	Bigint(8)	Primary key	Y
Au_username	Varchar(100)		Y
Au_personnelno	Varchar(18)		Y
Au_rs_pwd	Varchar(50)		Y
Au_email_id	Varchar(100)		Y
Au_usertype	Bigint(8)		
Au_strong_pwd	Varchar(50)		
Au_encryptedkey	Varchar(100)		
Au_defaultpage	Varchar(200)		Y
Au_ss_id	Int(4)		Y
Au_created_by	Bigint(8)		Y
Au_created_on	Datetime(8)		Y
Au_active_yn	Char(1)		Y

Table 3.18 Approles (Roles Information)

Field Name	Data Type	Key Reference	Not Null
Ar_id	Bigint(8)	Primary key	Y
Ar_name	Varchar(100)		Y
Ar_comment	Varchar(500)		
Ar_created_by	Bigint(8)		Y
Ar_created_on	Datetime(8)		Y
Ar_active_yn	Char(1)		Y

Table 3.19 AppRoleLinks (Roles against Links information)

Field Name	Data Type	Key Reference	Not Null
Arl_ar_id	Bigint(8)	Primary key , fk	Y
Arl_al_id	Bigint(8)	Primary key ,fk	Y
Arl_created_by	Bigint(8)		Y
Arl-created_on	Datetime(8)		Y
Arl_active_yn	Char(1)		Y

Table 3.20 AppUserRoles (Users against Roles Information)

Field Name	Data Type	Key Reference	Not Null
Aur_au_id	Bigint(8)	Primary key , fk	Y
Aur_ar_id	Bigint(8)	Primary key , fk	Y
Aur_created_by	Bigint(8)		Y
Aur_created_on	Datetime(8)		Y
Aur_active_yn	Char(1)		Y

3.4 INPUT DESIGN

The input to the ERS system is done through GUI screens. A lot many data that are options are maintained with drop downs so that the user has the least probability of making mistakes while entering the data. The screens follow a uniform design pattern which gives the user a uniform look and feel throughout. The screens also follow the logical flow of activities in the system.

A tree control is provided for the user to browse easily through the whole system. The tree control also lets the user clearly know where he / she is currently in the system. The tree control is displayed in a separate frame on the left side of the screen. The target of the links in tree control is displayed on the middle frame. A header and footer frame displays the username, the system name, signout and help controls, and version information.

All the search results will be shown in a data grid control. Hence datagrids with necessary columns are decided and placed on the respective screens. A lens image is provided for selecting information the users don't type into. They are displayed as a pop up on click of the icon and the user can select the information from the pop up window. An eraser control is also provided on the side to erase the information if needed. A date control is provided for the user to enter date in the fields. The date control helps users avoid mistakes in entering the date in various formats.

All the fields are placed in a sequential manner so that the user can enter data in one flow linearly.

3.5 OUTPUT DESIGN

Reports are generated as output for the users to view. Search criteria are specified so that the user can search for the details based on the information they know about the records. There are 4 basic reports produced by the ERS application.

- Demand Sheet
- Pipeline Report
- Position Comparison

The Demand Sheet report lets the user search details about the demands in the project for resources. The search criteria are Requisition from date, to date and service line. The status of each demand is displayed along with other details.

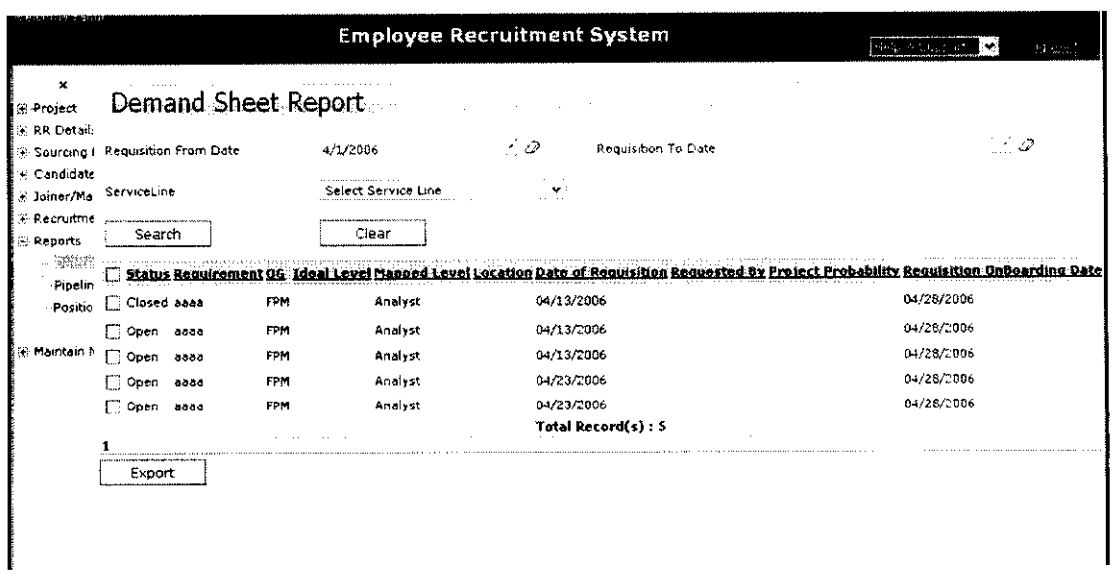


Figure 3.5 Demand sheet Report

The Pipe Line Report lets the user view details about the interviews including the status of each interview. The search criterion is service line. The candidates in various stages of the interview will be displayed.

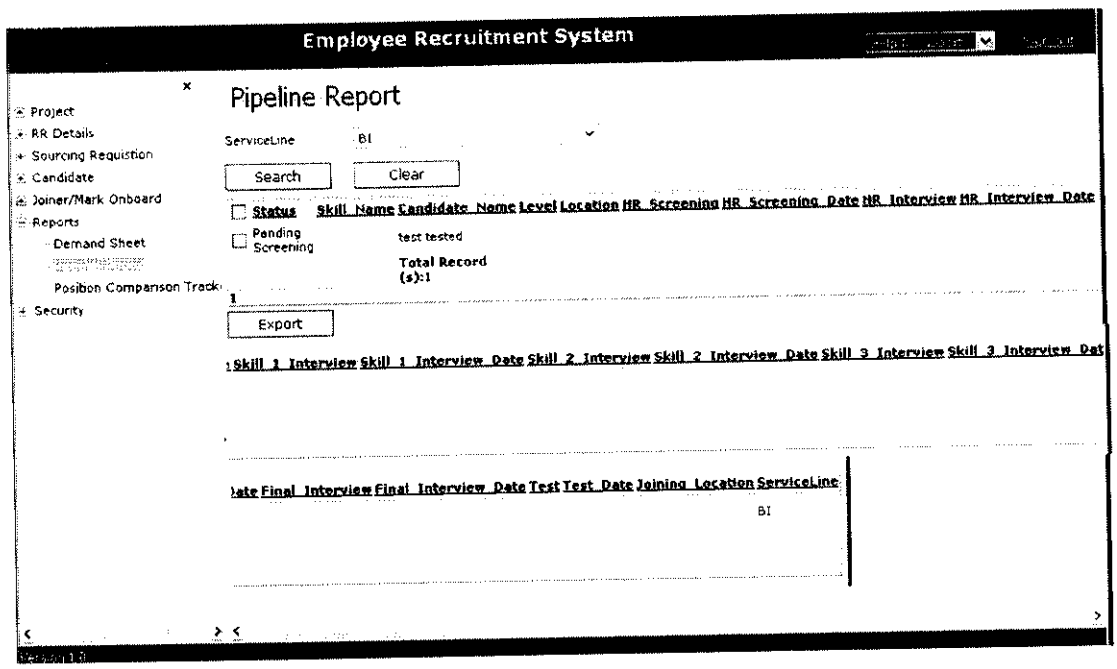


Figure 3.6 Pipeline Report

The Position Comparison Report shows details of candidates, such as education details, experience, organization details etc. The search criteria are organization, graduation and candidate status.

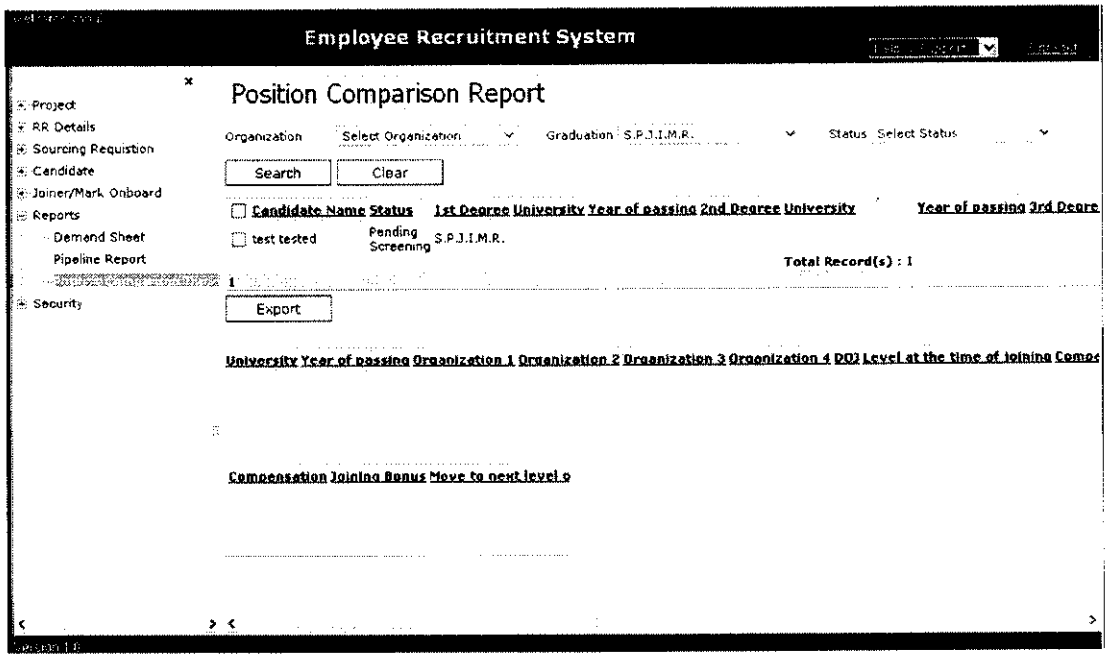
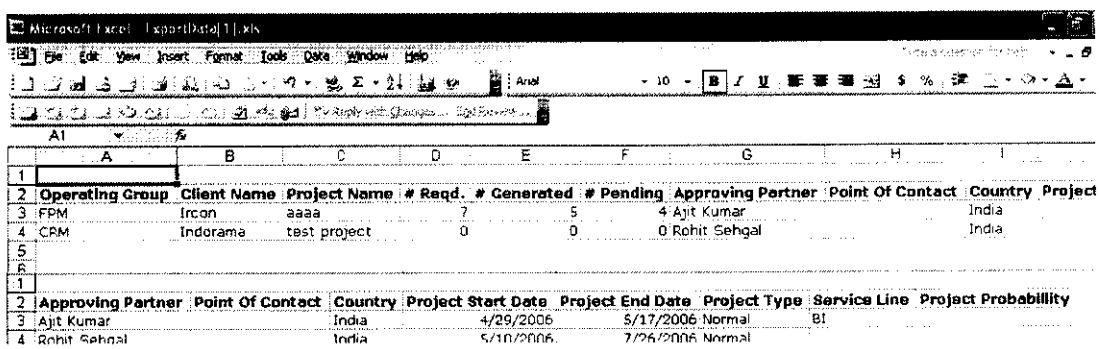


Figure 3.7 Position Comparison Report

The Search results also allows user to export the data to an excel sheet. A sample excel output for the Search Project screen is shown below



Operating Group	Client Name	Project Name	# Reqd.	# Generated	# Pending	Approving Partner	Point Of Contact	Country	Project
FPM	Ircon	aaaa	7	5	4	Ajit Kumar		India	
CRM	Indorama	test project	0	0	0	Rohit Sehgal		India	

Approving Partner	Point Of Contact	Country	Project Start Date	Project End Date	Project Type	Service Line	Project Probability
Ajit Kumar		India	4/29/2006	5/17/2006	Normal	BI	
Rohit Sehgal		India	5/10/2006	7/26/2006	Normal		

Figure 3.8 Excel output for search project screen.

CHAPTER 4

IMPLEMENTATION

System Implementation is the part of the software engineering life cycle, where, the design artifacts are converted to a working application. Coding is done in this stage using an apt framework and programming language, which would solve the specific problem the best way. Once the design is coded into a working application, it has to be verified, validated and tested in detail. The tested product if successful is deployed in the user environment.

4.1 TECHNOLOGY OVERVIEW

Microsoft .Net framework is used in constructing this application. VB.Net along with ASP.Net is used in the front end in coding the application. These languages provide a wide variety of options in implementing the need. ASP.Net is an ideal choice since the application is web based.

The ERS application implements its security in two levels. A user with ordinary password can only access data that is viewed generally. Sensitive data needs a Strong Password to view or edit. A user with strong password can view, edit or delete sensitive data. The other users see an encrypted value in these fields. Only a user with strong password can create the organization details for a candidate. The offer details that are sensitive such as compensation information can only be viewed by user with strong password. The passwords

are encrypted and stored. However these cannot be decrypted. Rather the password entered is encrypted and compared with the encrypted version in the database. This ensures security for sensitive data.

The security for users can be set from the application. A new user can have page level rights. The administrator can create users and assign the pages that are accessible for that user. The rest of the pages remain disabled for the user. For Eg, a project manager is given rights in project, RRD and SR pages and a recruiter is given rights to handle candidate recruitment pages.

Users are allowed to upload the project details, or candidate details directly to the database from an excel file. The template for such details is provided in those corresponding pages itself. The system also allows users to upload documents such as resume, education documents, certificates etc. ERS also allows user to export selective data from a data grid to an excel file.

The main operations are written as stored procedures. The stored procedures do all interaction with the database. These stored procedures are called from the visual basic code. Microsoft Sql Server is kept as the backend server.

4.2 SYSTEM VERIFICATION

System Verification answers the question "Am I building the product right?". It includes the review of interim work steps and interim deliverables during a project to ensure they are acceptable. Verification also determines if the system is consistent, adheres to standards, uses reliable techniques and prudent practices, and performs the selected functions in the correct manner. In

data access, it verifies whether the right data is being accessed, in terms of the right place and in the right way.

For eg, the drop downs gather data from the database, so each dropdowns should be verified whether they are bound to the correct database field. It is done during development of the key artifacts. Verification is a demonstration of consistency, completeness, and correctness of the software at each stage and between each stage of the development life cycle. In ERS, verification is done during the development itself. Each database bindings are verified after binding to test whether the control is bound to the right data field.

4.3 SYSTEM VALIDATION

Validation answers the question "Am I building the right product?". This checks whether the developer is moving towards the right product , whether the development is moving towards the actual intended product that was agreed upon in the beginning. Validation also determines if the system complies with the requirements and performs functions for which it is intended and meets the organization's goals and user needs. It is traditional and is performed at the end of the project. In data access , it checks whether we are accessing the right data, in terms of data required to satisfy the requirement.

Validation is performed after a work product is produced against established criteria ensuring that the product integrates correctly into the environment. It determines the correctness of the final software product by a development project with respect to the user needs and requirements.

Functional validation is done in ERS to check whether each of the functions is done correctly as expected in every page. Each control in every

page is designed to do some function. These functions are validated against whether they are done as the requirement states it. For eg, clicking a submit button should take the corresponding action such as saving the details in a database. Clicking the date control icon should pop up a date picker. This level of validation can continue to all the controls in the system. This checking is usually done after the system is developed so that all activities that are affected can be checked.

Field level validation is done in ERS to check whether each of the fields either accepts the data as expected and do the client side validation of data entered. For eg, a field level validation on a text box would check against proper type of data is entered and follows every rules such as length of entry, data type etc. Client side validation codes are written using various ASP.NET web controls. It provides five built-in validation controls – Regular expression, Required field, Range validator, comparison and custom validator.

The validation is done in a step by step process. First the page is loaded with the routines called from the validation controls. When the user moves between controls on a client-side validated form after changing a control's value, the validation events for the control that lost the focus are fired and appropriate error messages (if any) are displayed. If the user generates a form submit on a client-side validated form, the entire form is evaluated for any validation controls that are not valid. If even one control is not valid, the form will not be submitted. When the form is posted back to the server through a postback event, it is evaluated for validity on the server. This occurs even if the form was already evaluated on the client. If any of the validation controls are found to be invalid through the server-side check, the page is redisplayed with the appropriate error messages included. If the form passes all validation on both the server and

client, the page processing sequence continues with the event that triggered the postback firing.

4.4 SYSTEM TESTING

Testing is a process of high importance because, it decides whether the product is reliable or not. Hence it is done with an aim to break the system, finding maximum number of errors. These errors can then be corrected, thereby increasing the reliability of the system.

Standard procedures are followed in testing the ERS . First test cases are generated for each screen. These test cases will cover every possibility which could result in both positive and negative results. These test plans are maintained for any further testing done on the system. The test plan stores information such as, the test script/input, expected output, actual output, comments and the name of the tester. This plan is followed in all the types of testing done in the system.

Three types of testing are done in major in ERS.

- Unit Testing
- Integration Testing
- System Testing

Unit testing focuses on the individual units of the system. In ERS each page is tested separately as a unit. Initially the flow of control and data through that page is checked. When considering a module as a unit, the flow of data and control through the whole module is tested. The result is stored in the test plan. In a page, each control is further tested in unit testing. The process is done in all the pages of the system. Once the errors are rectified, the testing procedure is

repeated with same test cases to ensure this hasn't produced new errors. Hence this is a continuous process.

Integration testing tests the process of integrating the various modules to form the completed system. ERS followed bottom-up integration testing. Modules from the bottom most level are taken up individually, tested, integrated, and again tested. For eg. The add project screen is tested, the search project screen is tested, then they are integrated together and again tested. The project added in the first page should be displayed properly when searched in the project search page. This indicates proper flow of information in the project module. The same procedure is followed in other modules in the same level at first. Then the upper level is taken into action. The flow of data through the whole module in the upper level is taken and executed. A change of data made in one screen should have reflected in all other screens.

This process is continued from the page level to module level, finally to the system level. In the final stage, the whole system is taken together and tested for integration. A change in one place should be reflected through out the system. Regression testing is done after each change made into the software. This tests if the change has affected any part of the ERS negatively after the change was made. The whole set of test cases need to be run again to do the regression testing.

Now that the individual units are tested, integrated modules are tested, what remains is the entire system testing. The system testing takes into consideration the hardware, and the software. That is, the ERS should be able to be run on the specified hardware for variety of cases. The ERS is tested against recovery from errors.

Security testing is important in system testing. The system in no way shall be accessible to unauthorized users. Testing is done to ensure that if a user tries

to enter a URL that takes directly to a page inside the system without logging in, corresponding messages are displayed. The ERS would redirect the user back to the login page in such a case.

Another security issue involves the sensitive data in the system. Tests are done to ensure that the data is seen only if the user has entered through strong password. ERS displays encrypted form in other cases. ERS also stores the sensitive data in encrypted form. The passwords are stored in the system using an irreversible encryption algorithm.

One more level of security is concerned with user rights. Each user is applied rights page wise. The pages he does not have access to are not displayed to him in ERS. The page access rights are checked in each individual page. Moreover the administrative page rights can be assigned only by a user who has administrative rights. Hence the system is thoroughly tested against all kind of security loop holes.

These three testing covers all parts of ERS. The testing strategy is followed with an intent to find errors. Corrections made then make the software stronger.

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENTS

5.1 REVIEW OF ERS

The Employee Recruitment System has been successful in implementing the demand supply management in a project. The system has transitioned from manual excel sheet entries into an automated user friendly view and entry of data. Now, the system takes a user through a sequential flow of activities from creating a demand to recruiting a candidate onboard the company.

Implementation of security in the system has provided a good idea of the various aspects in it and the range of knowledge available in it. Overall the system has indeed implemented a lot of concepts in vb.net and asp.net. SQL queries were written in stored procedures to effectively manipulate the data. The requirements have been effectively satisfied through the Employee Recruitment System.

5.2 FUTURE ENHANCEMENTS

The Employee Recruitment System could be expanded further to include some other functionality. Provisions can be included to automatically add candidate information by uploading the information received in the email ids configured. Now the details are mapped from an excel sheet. Recruitment expenses can be made to be recorded within the application itself in the future

versions. Future versions could also include facility to add options to dropdowns from the front end itself.

APPENDIX 1

The major screens in the system are shown next.

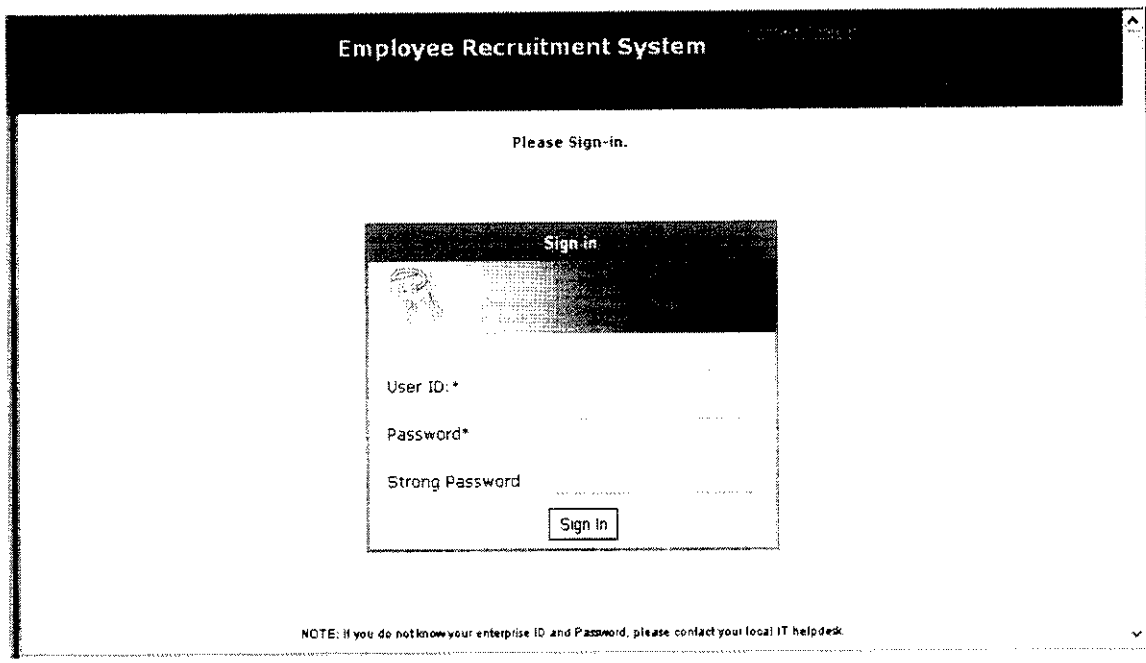


Figure A 1.1 Login Screen

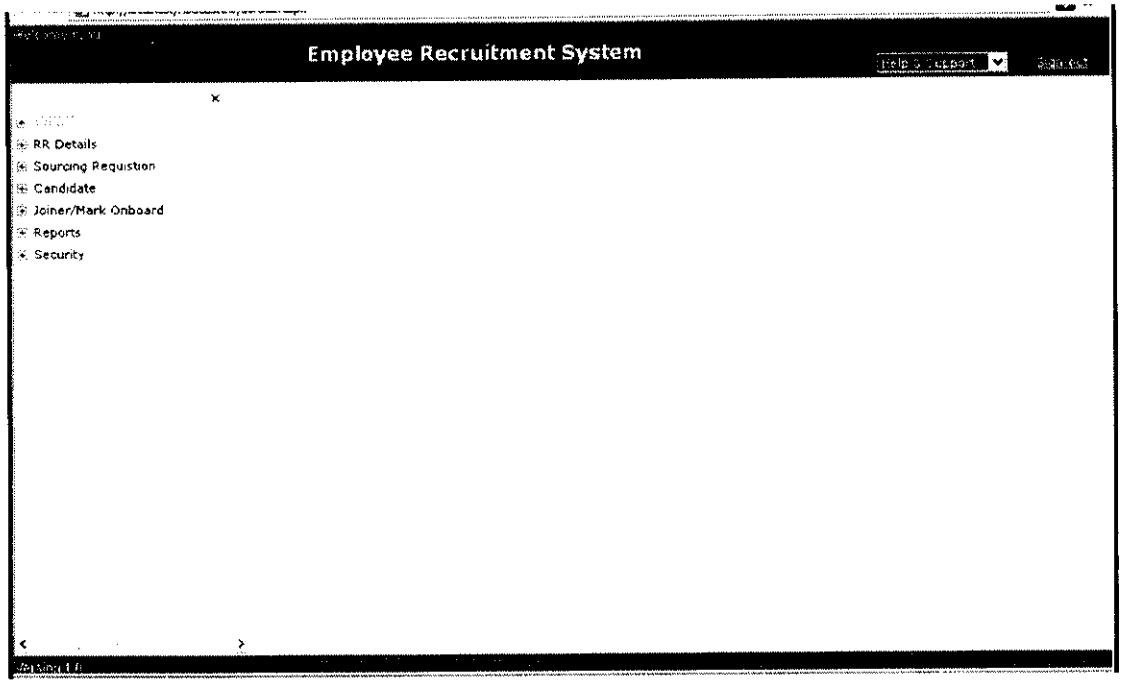


Figure A 1.2 Main Screen

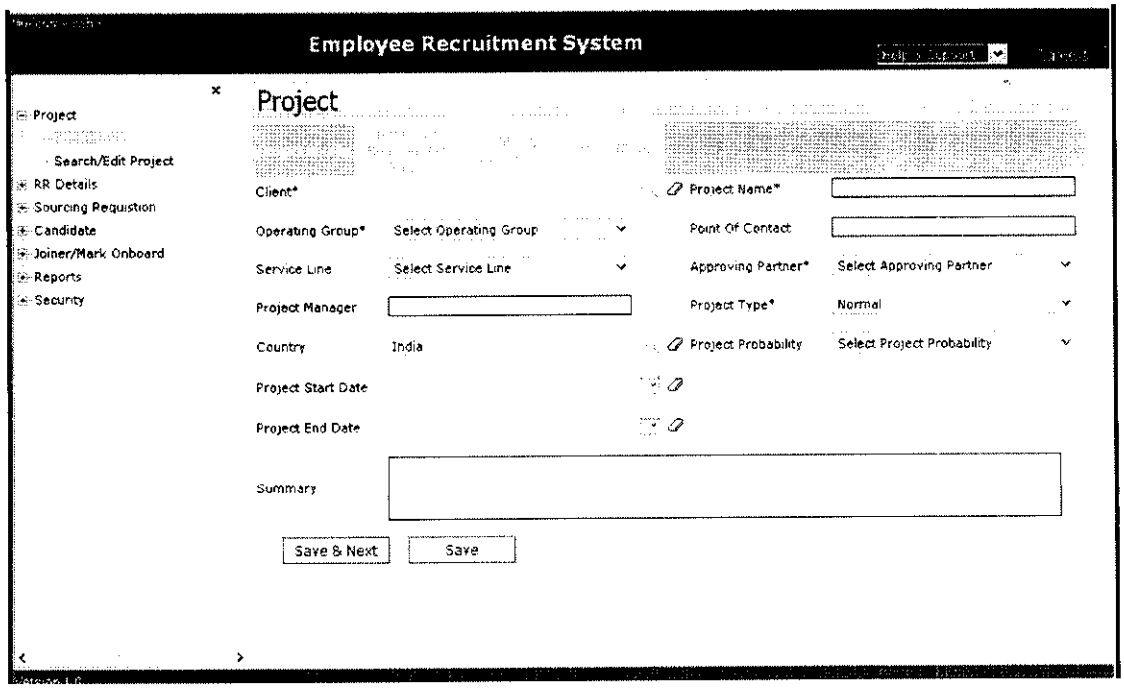


Figure A 1.3 Project -- Add

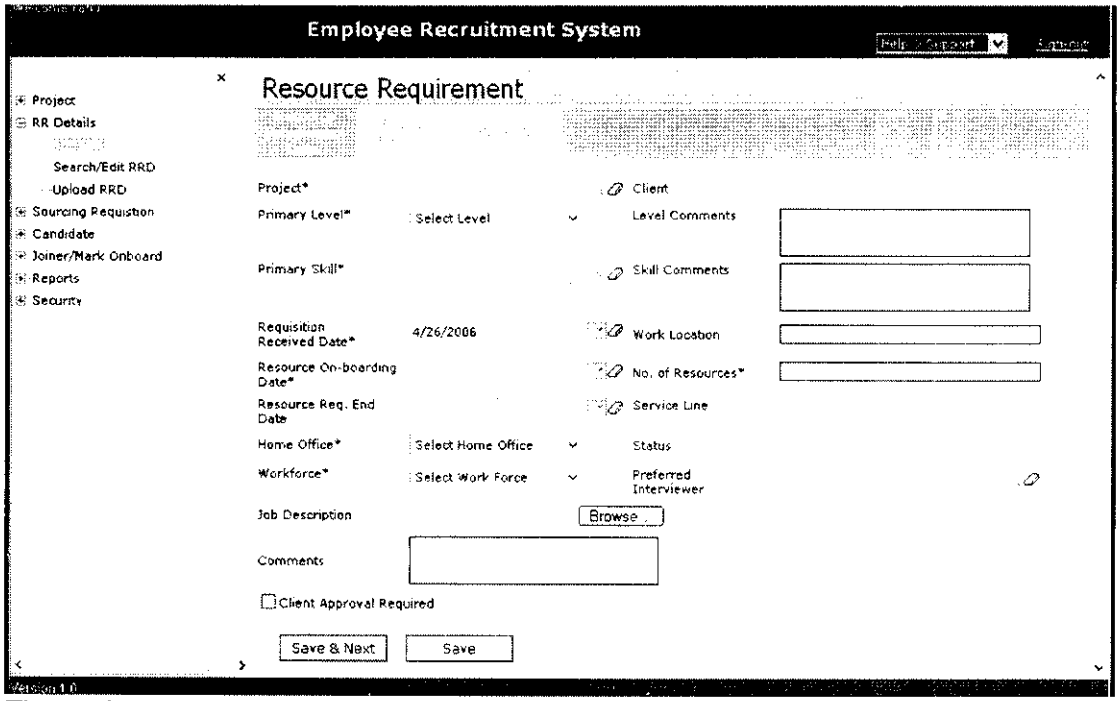


Figure A 1.4 Resource Requirement – Add

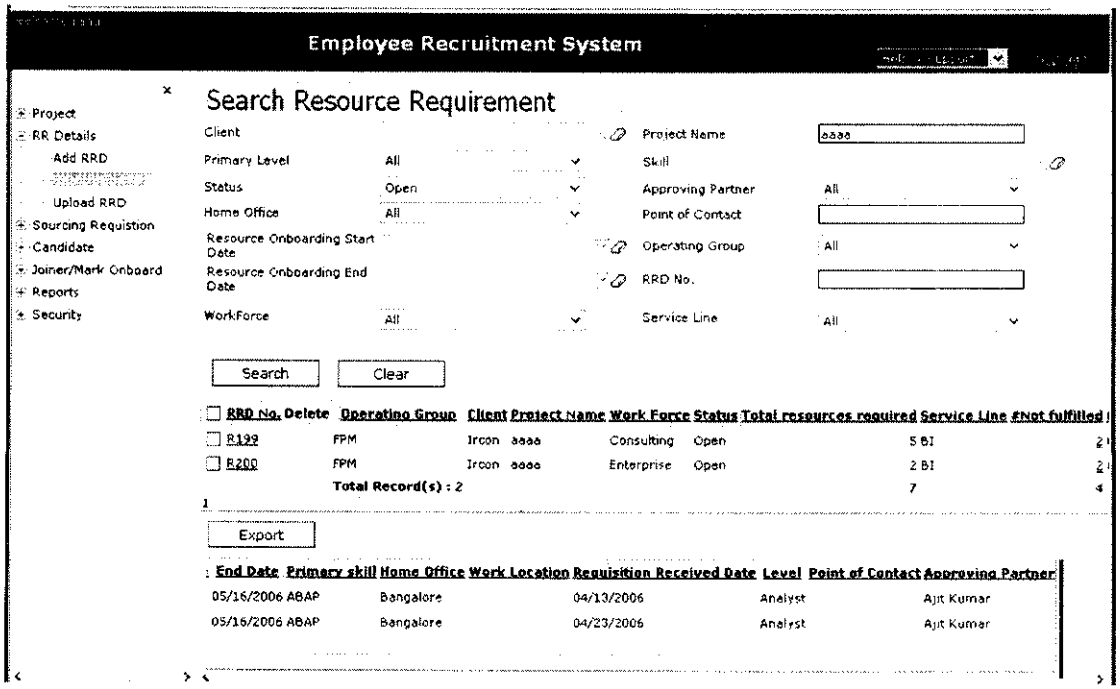


Figure A 1.5 Search Resource Requirement

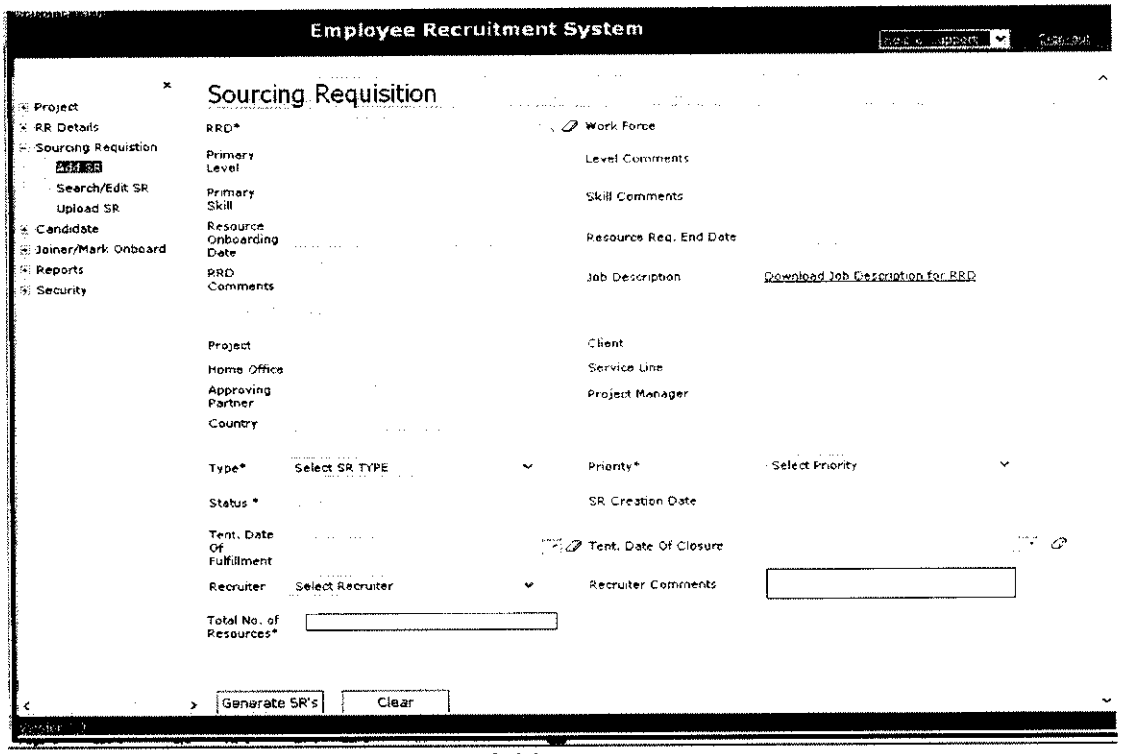


Figure A 1.6 Sourcing Requisition Add

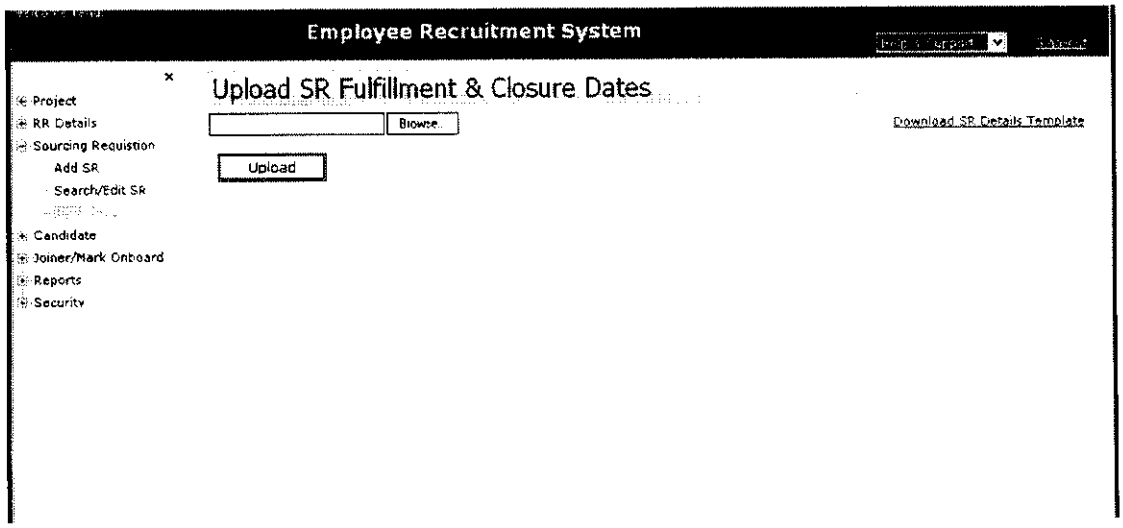


Figure A 1.7 Upload SR

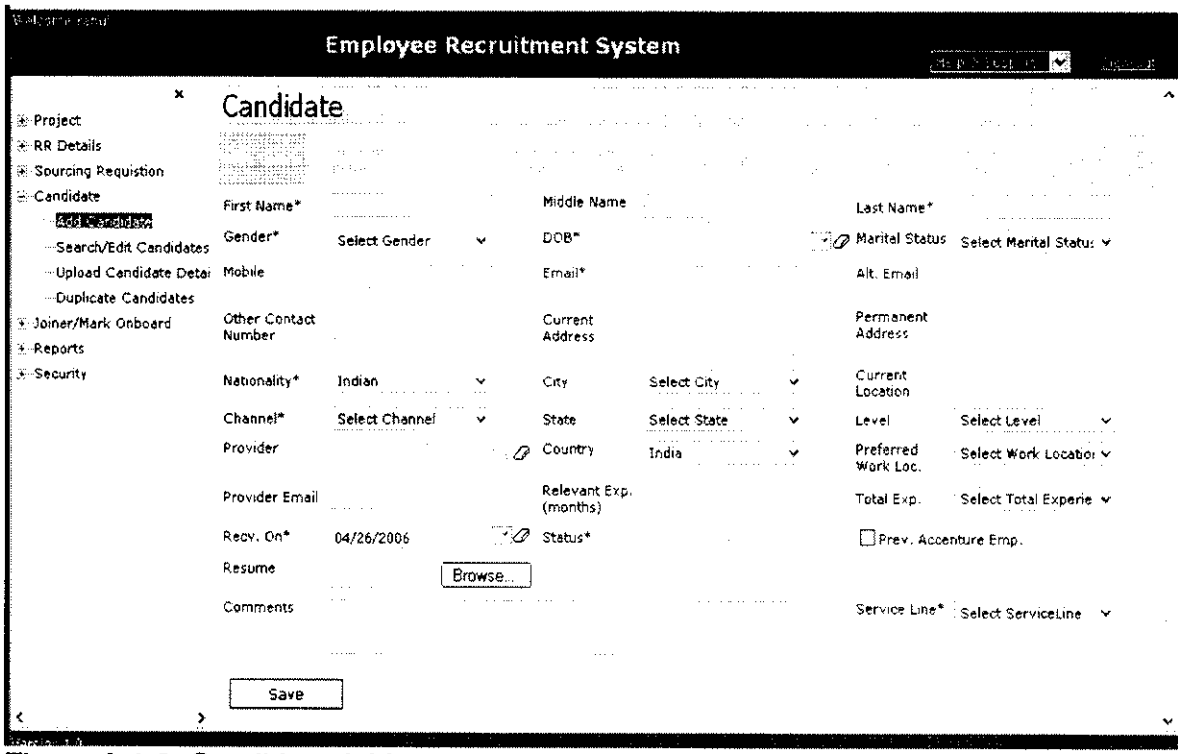


Figure A 1.8 Candidate Add

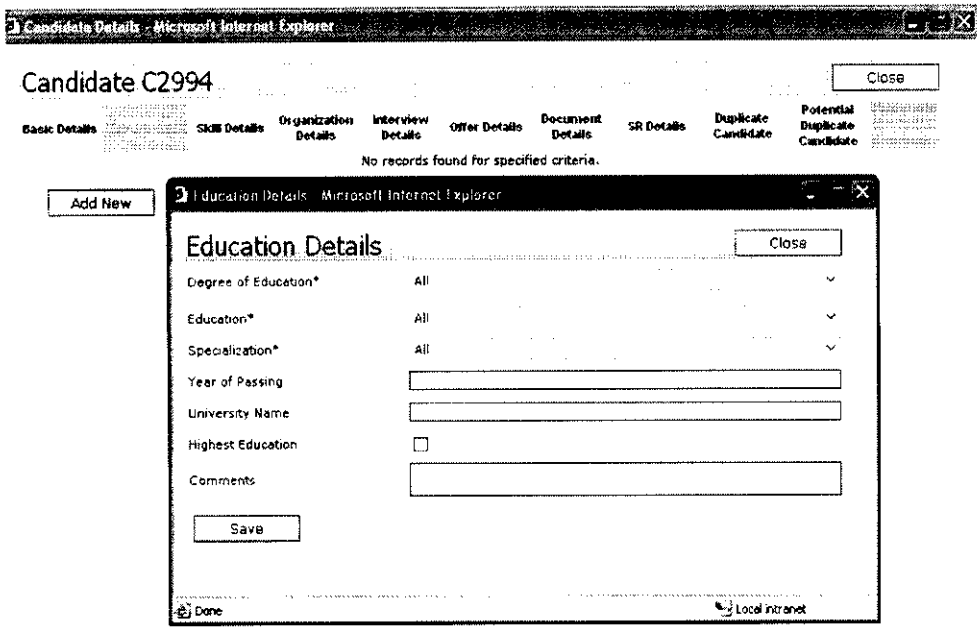


Figure A 1.9 Candidate -- Education Details Add

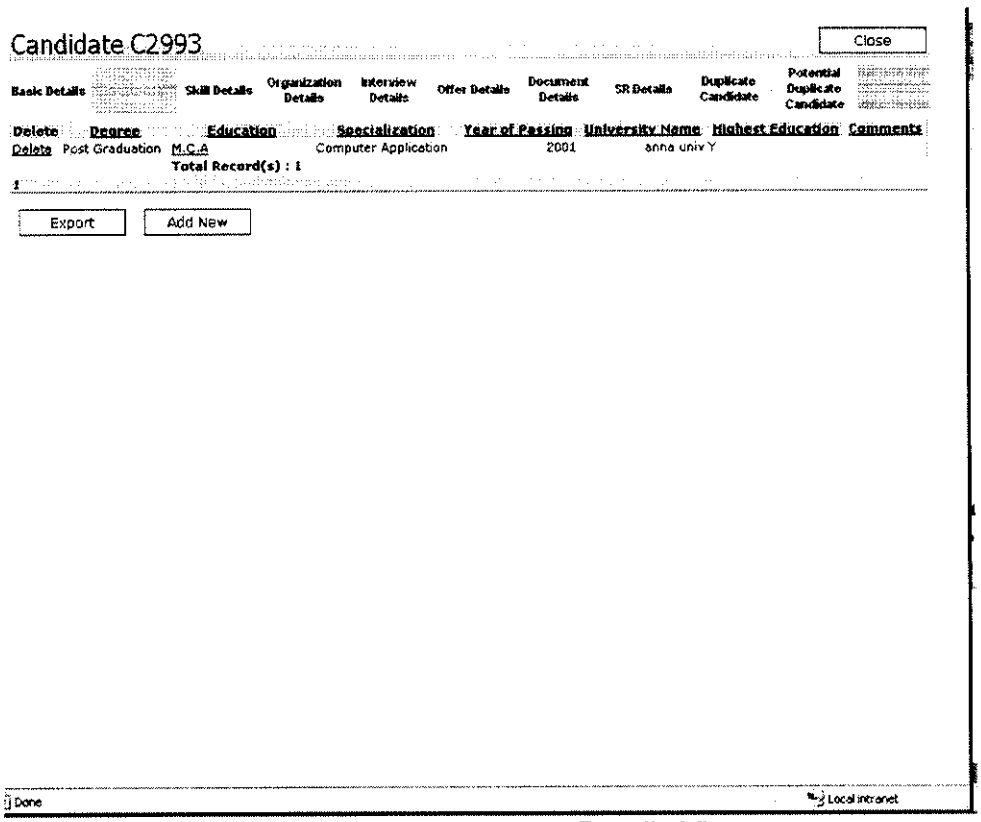


Figure A 1.10 Candidate Education Details View

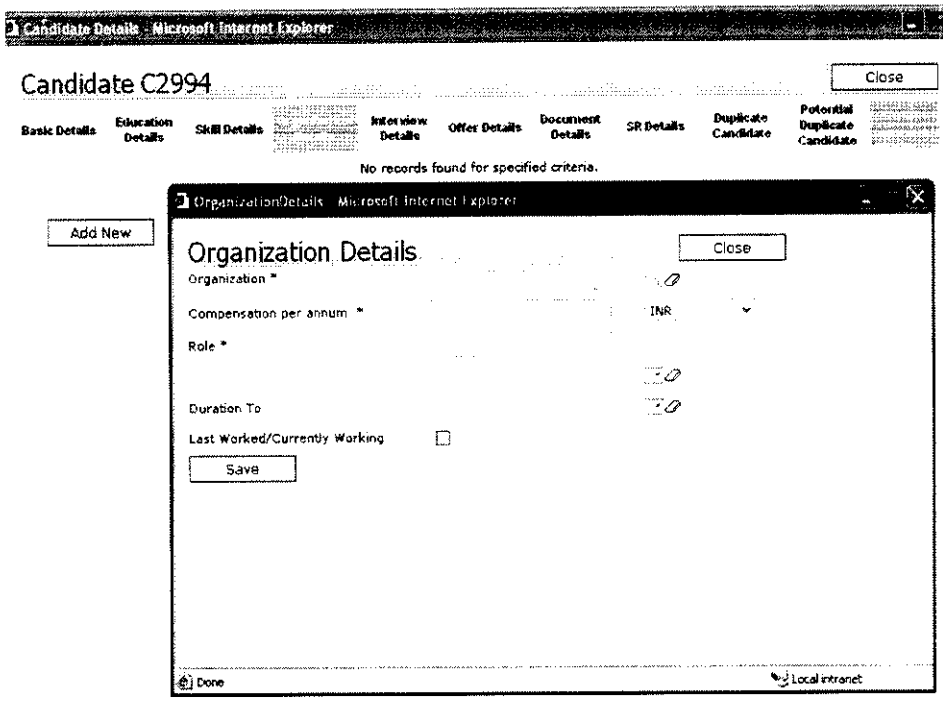


Figure A 1.11 Candidate Organization Details Add

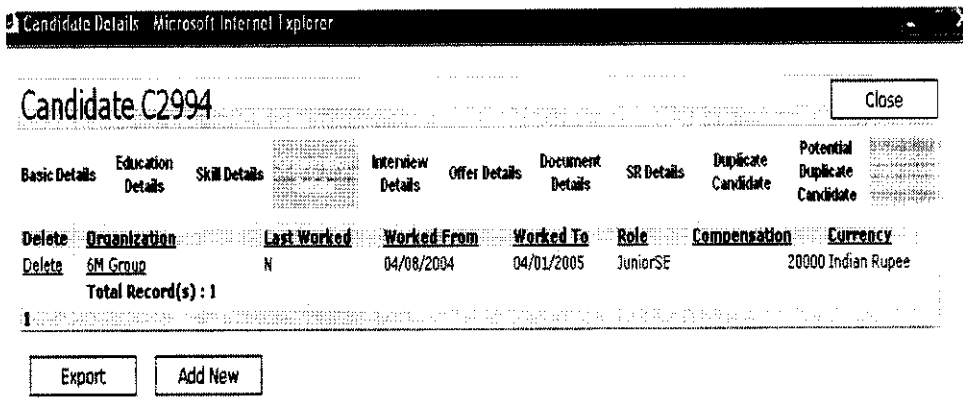


Figure A 1.12 Candidate Organization Details View

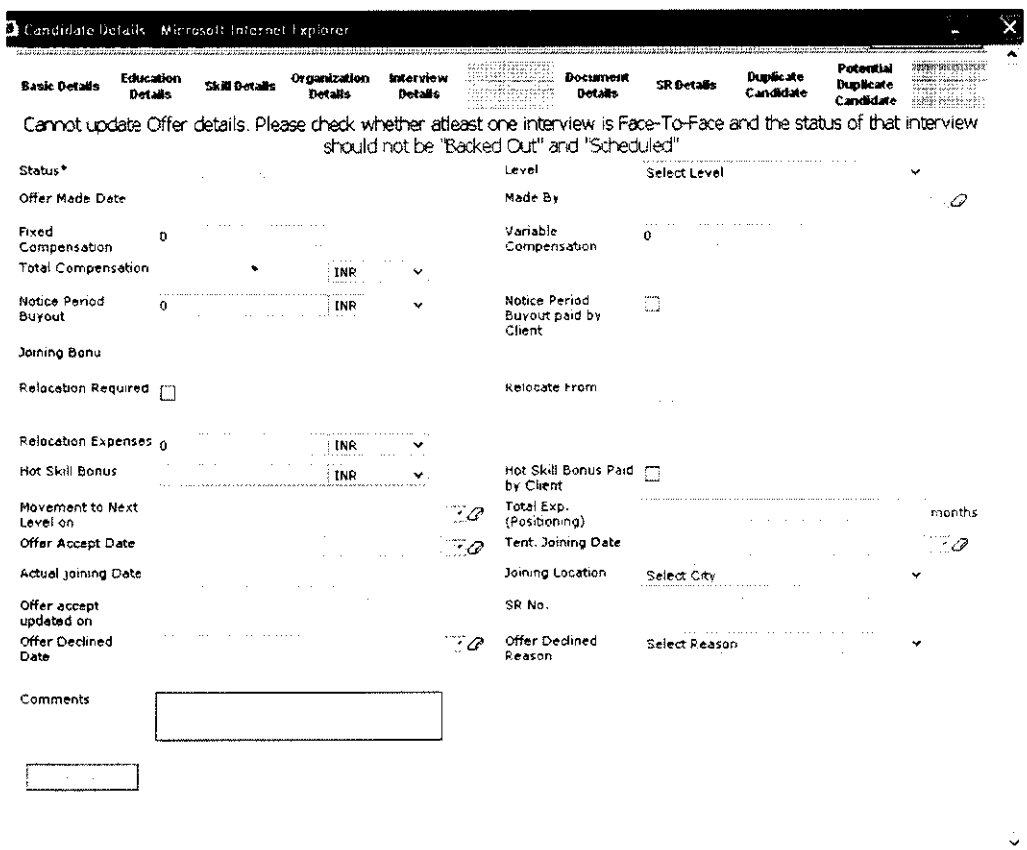


Figure A 1.13 Candidate Offer Details

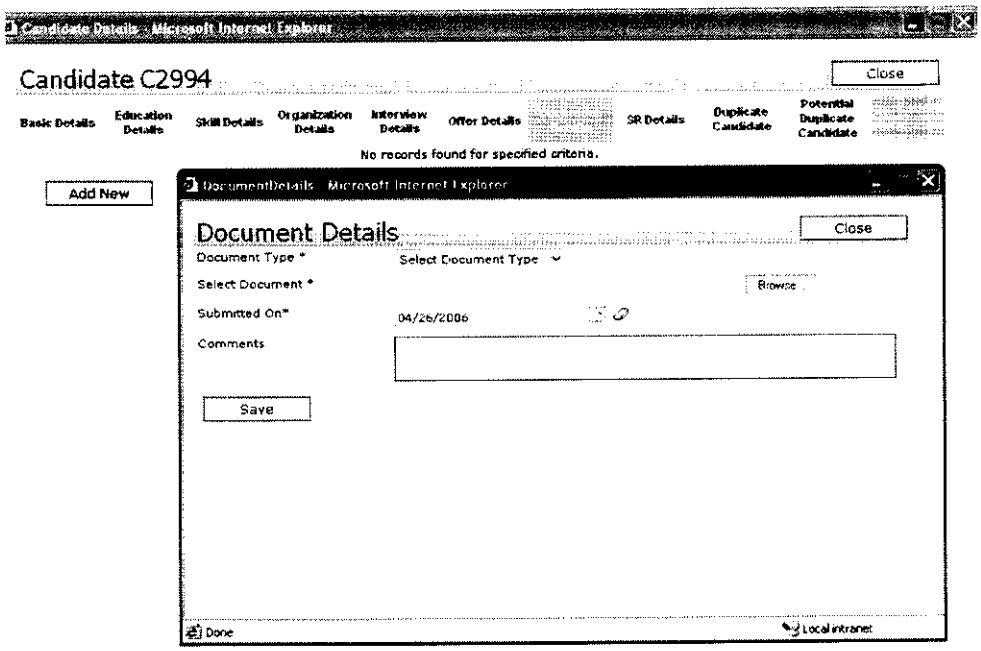


Figure A 1.14 Candidate Document Details Add

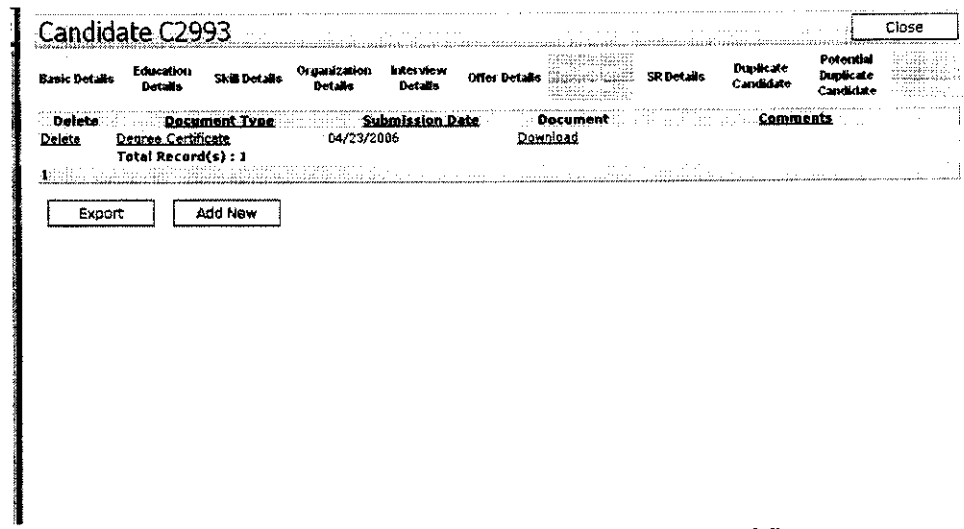


Figure A 1.15 Candidate Document Details View

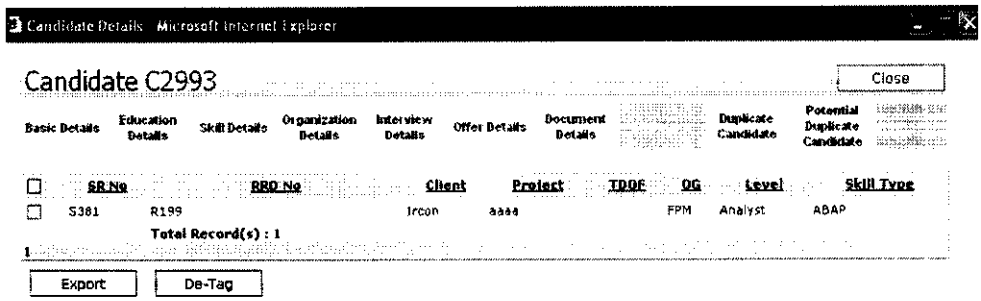


Figure A 1.16 Candidate SR Details

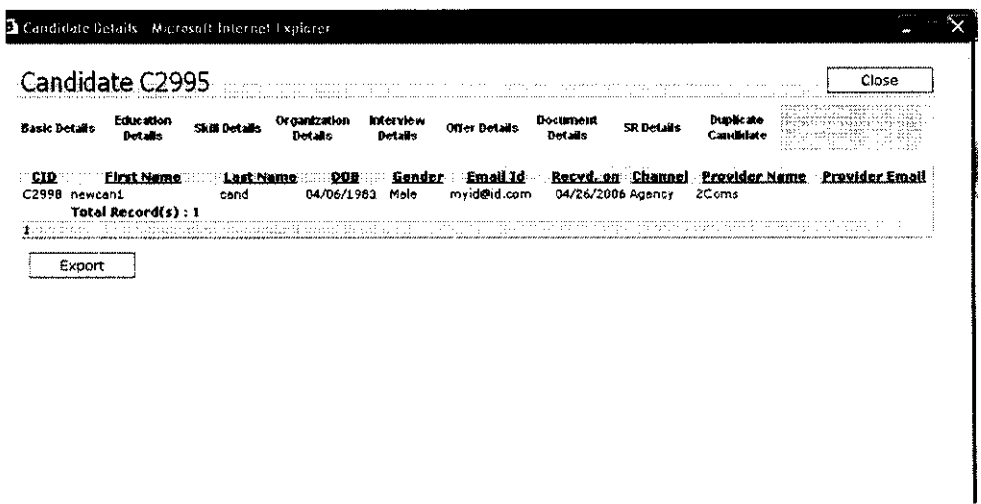


Figure A 1.17 Candidate Potential Duplicate

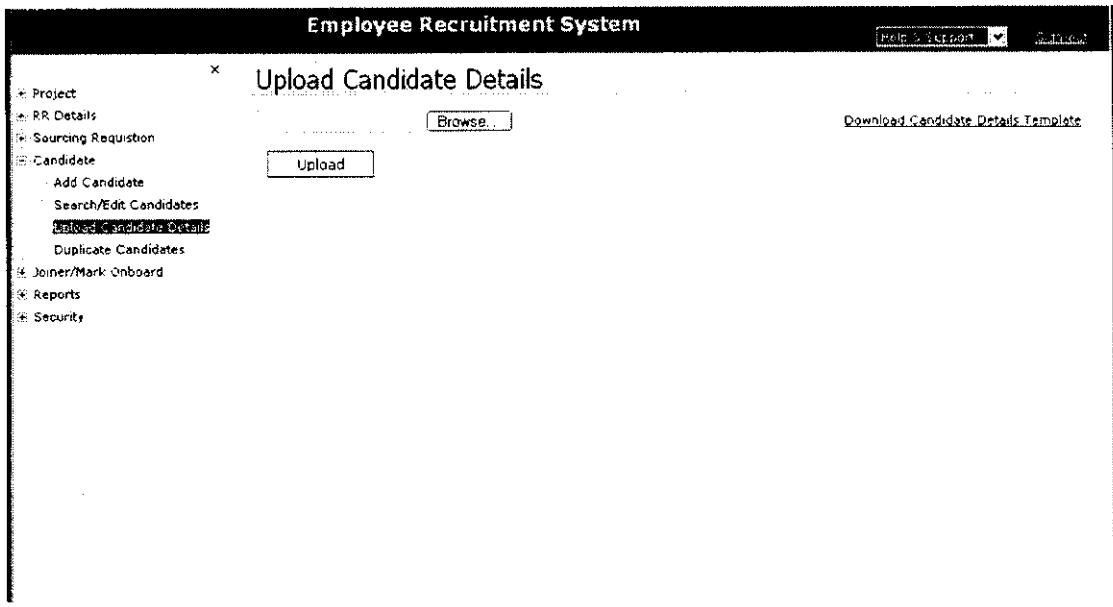


Figure A 1.18 Upload Candidate

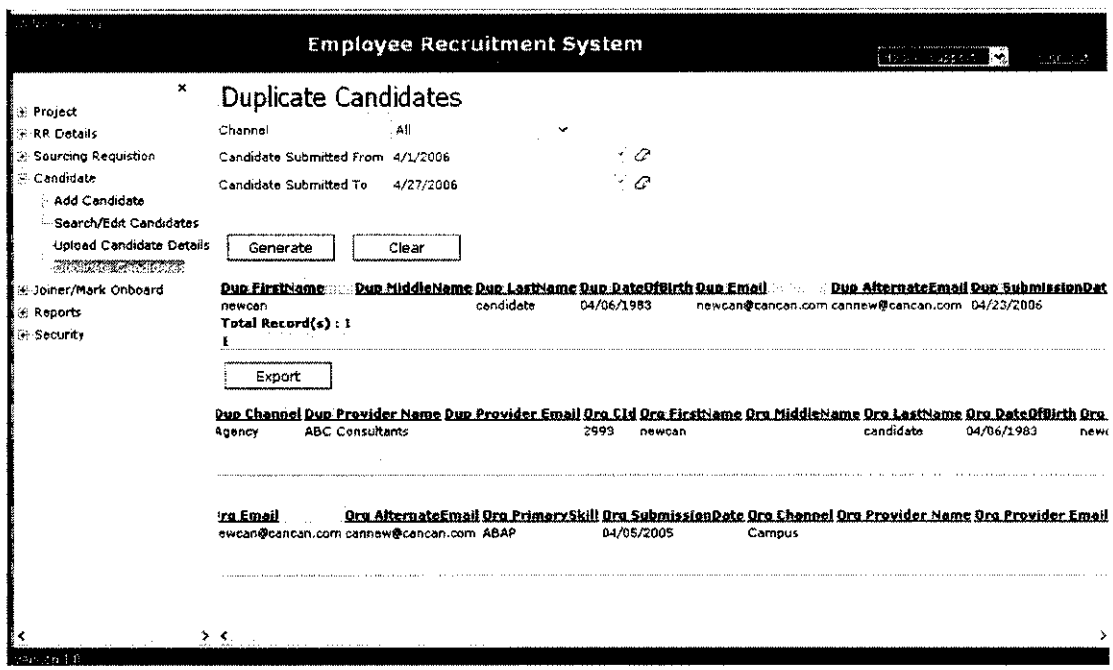


Figure A 1.19 Duplicate Candidate Search

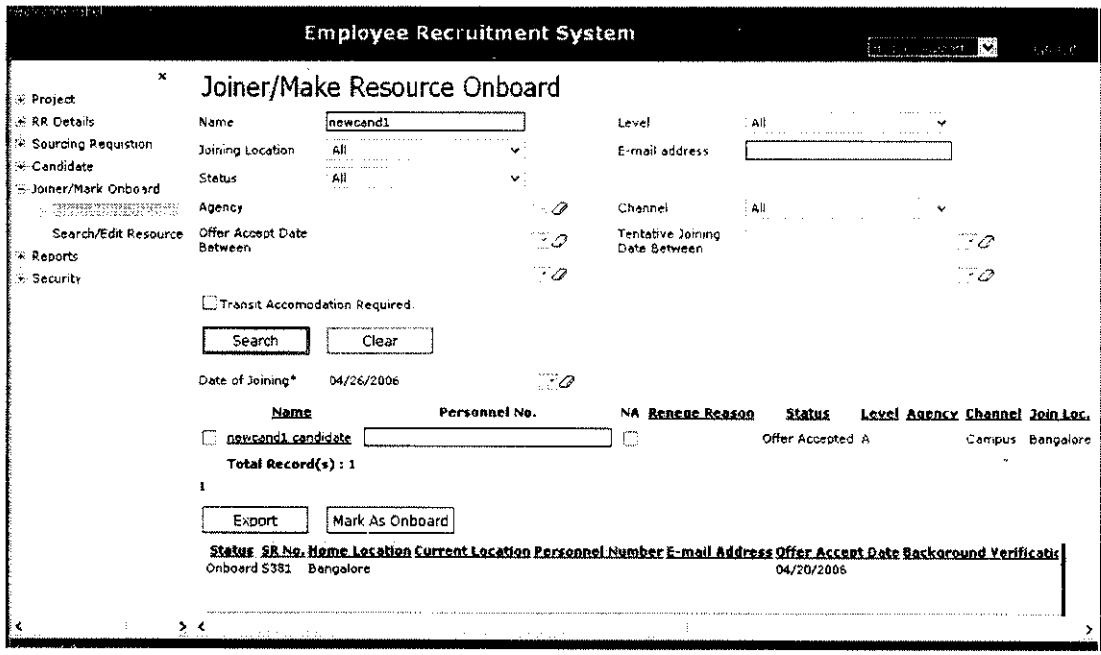


Figure A 1.20 Joiner/Resource Onboard

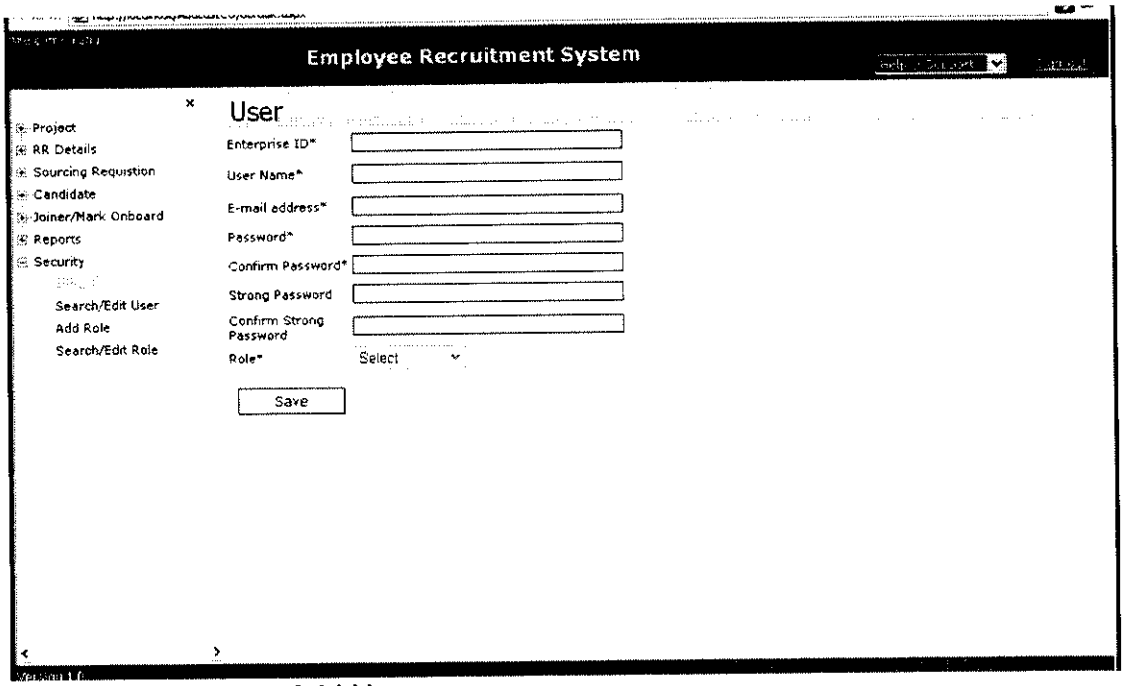


Figure A 1.21 Add User

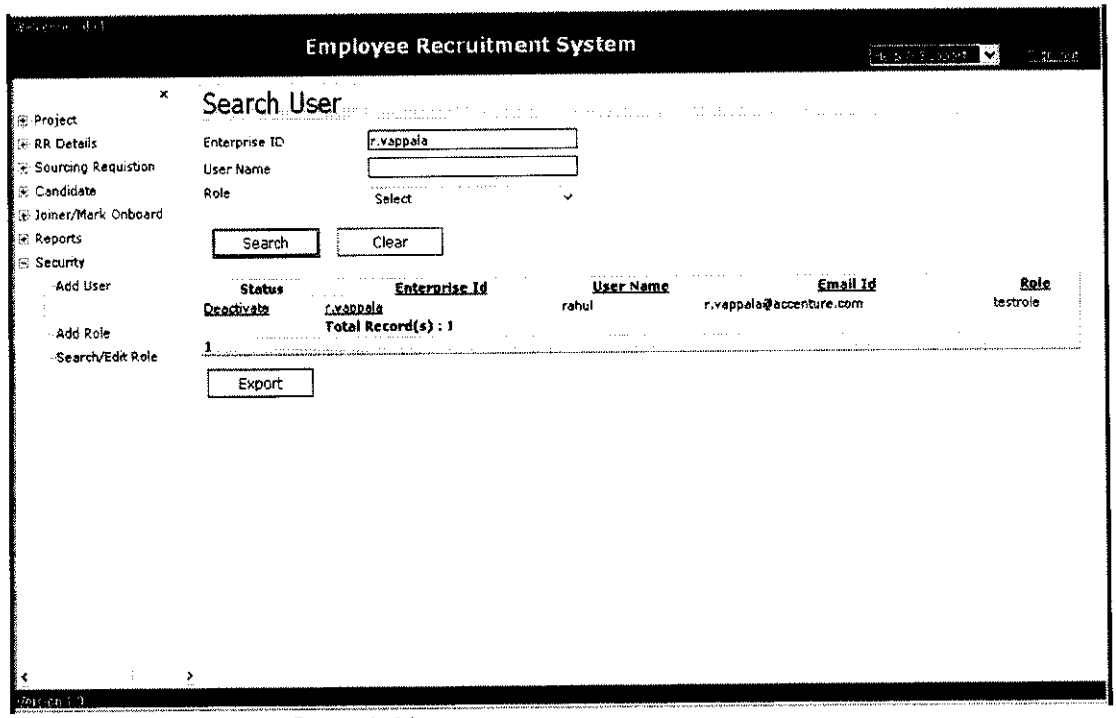


Figure A 1.22 Search User

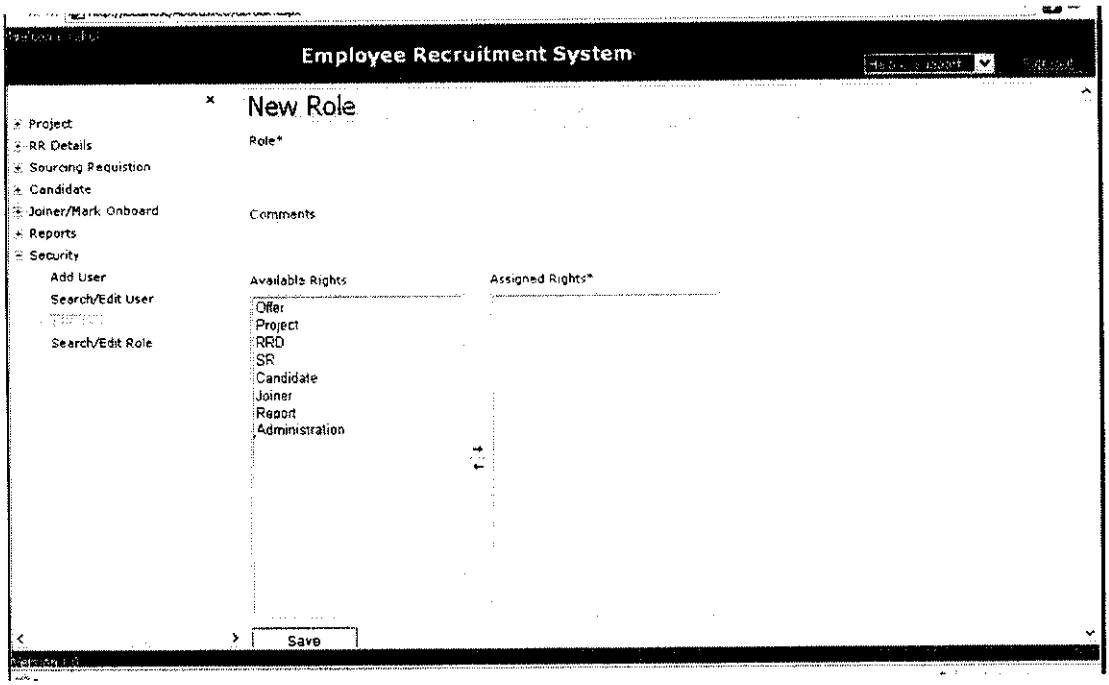


Figure A 1.23 Add Role

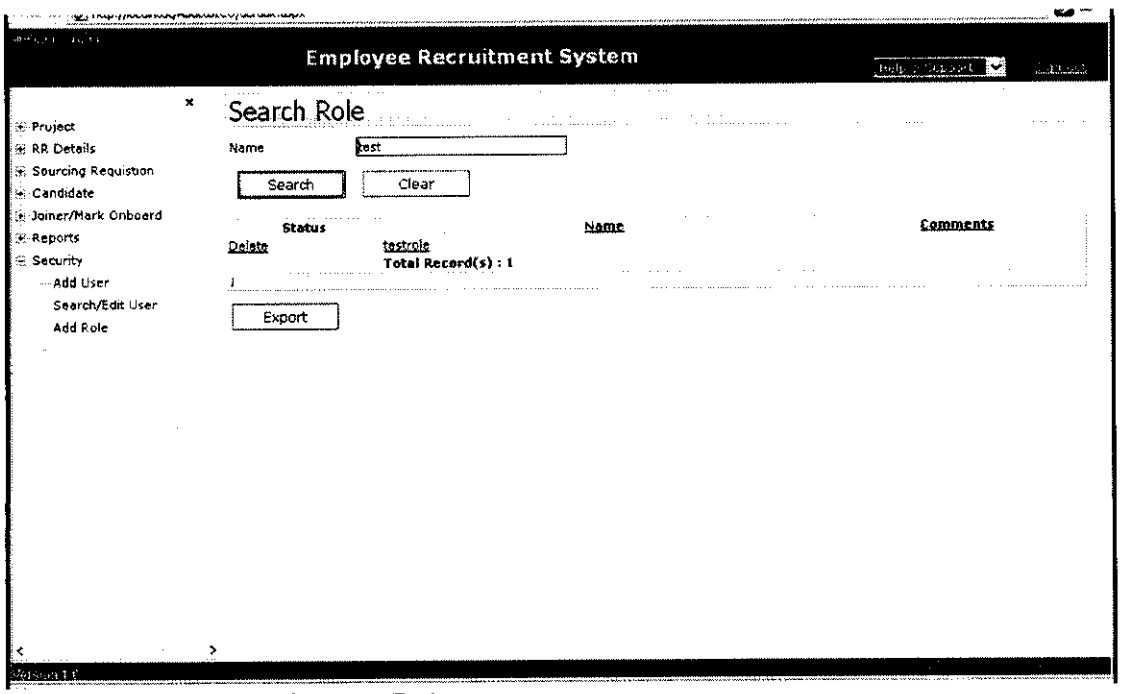


Figure A 1.24 Search Role

REFERENCE

1. Steven Holzner(2005), "Visual Basic .Net Programming Black Book", Dreamtech press

2. Roger S Pressman (1995), "Software Engineering – A Practitioners Approach" , TechMedia

3. Mesbah Ahmed, Chris Garett, Jeremy Faircloth, Chrish Payne, (2002), "ASP.NET Web Developers guide", Syngress Publishing.

4. James A. Senn (1989), "Analysis & Design of Information Systems", Second edition, McGraw-Hill International Edition.

5. http://msdn.microsoft.com/asp.net/

6. http://msdn.microsoft.com/library/