I. INTRODUCTION

E-CAMPUS is a Centralized Placement Exchange that is a web cum telephonic Application. It allows not only a job seeker to get latest and correct placement information, but it also helps job recruiters to get the correct list of candidates according to their criteria posted through java messaging services using the J2ME based technologies. As the name indicates that this is nothing but the online campus program which will be used by the many colleges, placement agencies and many more with advance messaging and alerts to the both recruiters and the job seekers about the latest information. This system mainly integrated using the following three key areas:

1) User application: - In which the each user comes and register with the system first, then got their unique userid & password for the further transactions with system. User can be of two types one is acts as recruiter and other is acts as the job seeker. During this users can submit their all the necessary details.

2) Messaging Application: - As in this case, this is used to create, send, receive and read messages using reliable, asynchronous and loosely coupled communication mechanism.

3) Alerts: - This is used to give the latest updates to the users by matching their criteria. This alert will be giving on their cell phones. (In our case we have to use the WAP gateway for the simulation.)

II. SYSTEM PHASES

This system is made up of the following phases:

1) Admin Phase
2) Job Seekers Phase
3) Recruiters Phase
4) Messages and Alert Phase

As explained above, Phase 2, 3 and 4 are exactly on the same line. While Admin section is related to the other management level tasks. Job Seekers that are non Students can directly send messages to Centralized Placement Exchange. Its Handling system fetches the messages, parses its, interprets the requirement using Natural Language Processing, generates and sends back a reply alert or the messages to the job seeker. In case of the last phase, the Admin, Job seeker or the recruiter can send read or delete the messages. Admin will send the latest alerts to the both job seekers and the recruiters in the system.

Key benefits of E-Campus System:

- **Improved Resource Utilization**: All aspects of resource and project management processes have been integrated into one standard web based solution. E-Campus System has provided accurate and detailed resource information to improve utilization.

- **Provided an automated and centralized method of tracking and monitoring based working.**

- **Improved Planning and Recharging**: The Commercial IT Division is now able to plan and manage projects more efficiently. The standardization of processes into one integrated business solution has radically improved project cost accuracy enabling the timely recharging and billing to internal customers. The result has been a dramatic increase in the number of chargeable hours, leading to accurate project costs.

- **Improved cash flow**: Many businesses are required to track the time spent on various projects/tasks. These are consulting companies or companies where there is a consulting component to the business. Timely billing to the client can significantly improve cash flow. A lot of time is spent collecting the billable time information from the consultants / employees.

- **Better project management**: There are businesses that collect time spent on various projects for better project management. Using E-Campus System will allow these businesses to collect information and improve project management.

- **Minimal roll out cost**: Apart from the business benefits E-Campus System, there are many operational benefits. Some of them are:
  - Low or virtually no training required.
  - Seamless integration into the Intranet.
  - The application is accessible worldwide

- **Real-Time Management Reporting**: E-Campus System web based reporting technology has helped reduce administrative time spent on management reporting.

III. TECHNOLOGY DESCRIPTION

- Java in Windows/Linux
- Java
- Jcreator
- Servlets
- Java Server Pages(JSP)
- Microsoft SQL Server 2000
- HTML

http://ijesc.org/
1) Java Technology: Java technology is both a programming language and a platform.

Why Java?

The Java programming language is a high-level language it provides following features which support development of web application

Java is simple:

Java uses automatic memory allocation and garbage collection. There are many programmers who can understand and write code in Java, so that many people can participate in developing open source software.

Java is object-oriented:

Object-oriented feature allows creating modular programs and reusable code. So these web services can be reused.

Java is platform-independent:

One of the most significant advantages of Java is its ability to move easily from one computer system to another. The ability to run the same program on many different systems is crucial to World Wide Web software, and Java succeeds at this by being platform-independent at both the source and binary levels.

Java is distributed:

Distributed computing involves several computers on a network working together. Java is designed to make distributed computing easy with the networking capability that is inherently integrated into it. Writing network programs in Java is like sending and receiving data to and from a file.

Java is interpreted:

An interpreter is needed in order to run Java programs. The programs are compiled into Java Virtual Machine code called byte code. The byte code is machine independent and is able to run on any machine that has a Java interpreter. With Java, the program need only be compiled once, and the byte code generated by the Java compiler can run on any platform

Java is secure:

Java is one of the first programming languages to consider security as part of its design. The Java language, compiler, interpreter, and runtime environment were each developed with security in mind.

Java is robust:

Robust means reliable and no programming language can really assure reliability. Java puts a lot of emphasis on early checking for possible errors, as Java compilers are able to detect many problems that would first show up during execution time in other languages.

Java is multithreaded:

Multithreaded is the capability for a program to perform several tasks simultaneously within a program. In Java, multithreaded programming has been smoothly integrated into it, while in other languages, operating system-specific procedures have to be called in order to enable multithreading. Multithreading is a necessity in visual and network programming.

Single-paradigm language:

Java is predominantly a single-paradigm language. In case of the many programming languages, we can either compile or interpret the program in order to run that program on computer successfully. But Java is both compiled and interpreted means any program is both compiled and interpreted. Thus to run any program we first translate that program into an intermediate language called Java byte codes using the java compiler (JAVAC). Java byte code is nothing but the platform-independent codes which is interpreted by the interpreter on the Java platform. Then next step is that java interpreter (JAVA) which parses and runs each Java byte code instruction on the computer. At the last compilation happens just once but the interpretation occurs each time when the program is executed. The following figure illustrates how this works.

![Java environment diagram]

Figure 1. Java environment

Java byte codes are as the machine code and the nothing but the instructions for the Java Virtual Machine (Java VM). Every Java interpreter is an implementation of the java virtual machine (JVM) even if that java interpreter is a development tool or a Web browser that can run applets. Thus this makes the java as the platform independent as we can compile our program into bytecodes on any platform that has with a Java compiler, then in turn bytecodes can then be run on any
implementation of the Java VM. Hence the Java’s motto comes here that “Write once, Run anywhere”. Following figure shows the above scenario.

**Figure 2. Java platform independent**

2) **Java IDEs/J-Creator**
   All programs you write will be written in an IDE, or integrated development environment. This just means the visual interface where you write your programs. They also rely on the Java Platform you have installed to help you to compile, run, and debug your code. There are many different choices of IDEs, and all of them are perfectly acceptable. Your choice of IDE is based on personal preference. Some popular IDEs include NetBeans IDE from Sun Microsystems, JCreator, and Eclipse. All Java tutorials written in this web site will be using the Eclipse IDE. Manner and help them realize the promise of information anytime, anywhere, on any device.

3) **Java Servlets:**
   It is nothing but the server side java program which provides the additional services to the server. Thus we can also say that small program that runs on a server. Servlet is usually refers to a Java applet which is runs within a Web server environment. Java servlets are as an alternative to CGI programs that were used most widely. The main difference between the servlets and the CGI is that a Java servlet is persistent means that once it is started, it stays in memory and can fulfill multiple requests. But in case of the CGI, a CGI program disappears once it has fulfilled a request.

**Advantages of Java Servlets**

1) **Portability:**
   As we know that the servlets are written in java and follow well known standardized APIs so they are highly portable across operating systems and server implementations. We can develop a servlet on Windows machine running the tomcat server or any other server and later we can deploy that servlet effortlessly on any other operating system like UNIX server running on the iplanet/Netscape Application server. So servlets are writing once and run anywhere (WORA) program.

2) **Efficiency:**
   As compared to CGI the servlets invocation is highly efficient. When the servlet get loaded in the server, it remains in the server's memory as a single object instance. However with servlets there are N threads but only a single copy of the servlet class. Multiple concurrent requests are handled by separate threads so we can say that the servlets are highly scalable.

3) **Safety:**
   As servlets are written in java, servlets inherit the strong type safety of java language. Java's automatic garbage collection and a lack of pointers mean that servlets are generally safe from memory management problems. In servlets we can easily handle the errors due to Java's exception handling mechanisms. If any exception occurs then it will throw an exception.

4) **Integration**
   Servlets are tightly integrated with the server. Servlet can use the server to translate the file paths, perform logging, check authorization, and MIME type mapping etc.

5) **Extensibility**
   The servlet API is designed in such a way that it can be easily extensible. As it stands today, the servlet API support Http Servlets, but in later date it can be extended for another type of servlets.

6) **Inexpensive**
   there are number of free web servers available for personal use or for commercial purpose. Web servers are relatively expensive. So by using the free available web servers you can add servlet support to it.

7) **About JSP**
   Java Server Pages (JSP) is a technology based on the Java language and enables the development of dynamic web sites.
JSP was developed by Sun Microsystems to allow server side development. JSP files are HTML files with special Tags containing Java source code that provide the dynamic content. The following shows the Typical Web server, different clients connecting via the Internet to a Web server. In this example, the Web server is running on UNIX and is the very popular Apache Web server.

First static web pages were displayed. Typically these were people’s first experience with making web pages so consisted of My Home Page sites and Company marketing information. Afterwards Perl and C were languages used on the web server to provide dynamic content. Soon most languages including Visualbasic, Delphi,C and Java could be used to write applications that provided dynamic content using data from text files or database requests. These were known as CGI server side applications. ASP was developed by Microsoft to allow HTML developers to easily provide dynamic content supported as standard by Microsoft’s free Web Server, Internet Information Server (IIS). JSP is the equivalent from Sun Microsystems, a comparison of ASP and JSP will be presented in the following section. The following diagram shows a web server that supports JSP files. Notice that the web server also is connected to a database.

**Figure. 3. Client-Server architecture.**

**Figure. 4. Java client-server architecture-II**
JSP source code runs on the web server in the JSP Servlet Engine. The JSP Servlet engine dynamically generates the HTML and sends the HTML output to the client’s web browser.

**Why use JSP?**
JSP is easy to learn and allows developers to quickly produce web sites and applications in an open and standard way. JSP is based on Java, an object-oriented language. JSP offers a robust platform for web development.

**Main reasons to use JSP:**
Multi platform Component reuse by using JavaBeans and EJB.

**Advantages of Java.** You can take one JSP file and move it to another platform, web server or JSP Servlet engine.

![Figure 5: Java client-server architecture with platform independence](http://ijesc.org/)

This means you are never locked into one vendor or platform. HTML and graphics displayed on the web browser are classed as the presentation layer. The Java code (JSP) on the server is classed as the implementation. By having a separation of presentation and implementation, web designer’s work only on the presentation and Java developers concentrate on implementing the application.

**JSP compared to ASP**
JSP and ASP are fairly similar in the functionality that they provide. JSP may have slightly higher learning curve. Both allow embedded code in an HTML page, session variables and database access and manipulation. Whereas ASP is mostly found on Microsoft platforms i.e. NT, JSP can operate on any platform that conforms to the J2EE specification. JSP allow component reuse by using JavaBeans and EJBs. ASP provides the use of COM / ActiveX controls.

**JSP compared to ASP.NET**
ASP.NET is based on the Microsoft .NET framework. The .NET framework allows applications to be developed using different programming languages such as Visual Basic# and JavaScript. JSP and Java still has the advantage that it is supported on many different platforms and the Java community has many years of experience in designing and developing Enterprise quality scalable applications. This is not to say that ASP.NET is bad, actually it is quite an improvement over the old ASP code.

**JSP compared to Servlets**
A Servlet is a Java class that provides special server side service. It is hard work to write HTML code in Servlets. In Servlets you need to have lots of println statements to generate HTML. JSP pages are converted to Servlets so actually can do the same thing as old Java Servlets.

**JSP architecture:**
JSPs are built on top of SUN Microsystems’ servlet technology. JSPs are essential an HTML page with special JSP tags embedded. These JSP tags can contain Java code. The JSP file extension is .jsp rather than .htm or .html. The JSP engine parses the .jsp and creates a Java servlet source file. It then compiles the source file into a class file, this is done the first time and this why the JSP is probably slower the first time it is accessed. Any time after this the special compiled servlet is executed and is therefore returns faster.
Figure 6. JSP request and response processing

Steps required for a JSP request:

1. The user goes to a web site made using JSP. The user goes to a JSP page (ending with .jsp). The web browser makes the request via the Internet.
2. The JSP request gets sent to the Web server.
3. The Web server recognizes that the file required is special (.jsp), therefore passes the JSP file to the JSP Servlet Engine.
4. If the JSP file has been called the first time, the JSP file is parsed, otherwise go to step 7.
5. The next step is to generate a special Servlet from the JSP file. The entire HTML required is converted to printing statements.
6. The Servlet source code is compiled into a class.
7. The Servlet is instantiated, calling the init and service methods.
8. HTML from the Servlet output is sent via the Internet.
5) HTML results are displayed on the user's web browser.
6) About Ms-sqlserver2000:
7) This is the database backend tool which most suitable any other available back ends.

Why sql server 2000:
I preferred to use the SQL Server 2000 as the backend which has the following advantages over the other backend’s such as oracle 9i:

- SQL Server 2000 is very cheaper and feasible to buy than Oracle 9i Database.
- SQL Server 2000 always holds on giving the top TPC-C performance and price/performance results ratio.
- SQL Server 2000 is generally accepted as easier to install, use and manage. It is freely available to use and hence widely acceptable.

So in general, the Microsoft SQL Server 2000 is a full-featured relational database management system (RDBMS). This backend technique offers a variety of administrative tools that are used to make the easy and flexible database development, maintenance and administration tools. There are six main and most frequently used tools such as:

1) Enterprise Manager: - It is the main administrative tool for SQL Server installations and creating the new databases.
2) Query Analyzer: - Query Analyzer basically offers quick and direct methods which are used to perform queries against any of available SQL Server databases.
3) SQL Profiler: - It provides a window into the inner workings of your database. SQL Profiler allows any user to capture and replay system “traces” that log various activities.
4) Service Manager: - It is used to control the MSSQL Server (the main SQL Server process), MSDTC (Microsoft Distributed Transaction Coordinator) and SQL Server Agent processes.

Data Transformation Services: -this tool always provides the extremely flexible method for the importing and the exporting data

1) Between an ms-sql server installation and the different number of the other formats.
2) Books Online: -this is also the best facility of the Microsoft group which gives free access on the books which gives lot of the information about the sql server 2000 components and all which really helpful to the users.

6) About HTML
I organize the HTML information in following points.
1) Building Dynamic GUIs: -Now days, it is fashion that the great Web site not only has to look great but also it also has to be more user friendly and load fast. So the dynamic HTML is the solution to the problem.
2) HTML stands for Hyper Text Markup Language.
3) An HTML file is nothing but a text file which containing small markup tags.
4) The all available html markup tags tell that the Web browser how to display the page.
5) An every HTML file must have an htm or html file extension.
Software requirement specifications

In the following section we will go through the all the concepts that are related to the operating environment of the system:

1) OPERATING ENVIRONMENT

Development based on the customer requirements, technology capability and the skills available, the following technologies are implemented with the project.

Software specifications:

<table>
<thead>
<tr>
<th>Category (Software tools)</th>
<th>Software Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Microsoft Windows, Linux</td>
</tr>
<tr>
<td>Front End Development Tool</td>
<td>Eclipse</td>
</tr>
<tr>
<td>Front End JSP,Servlet,HTML,CSS</td>
<td></td>
</tr>
<tr>
<td>Back End Microsoft SQL Server 2000</td>
<td></td>
</tr>
</tbody>
</table>

Hardware specifications:

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Minimum Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microprocessor</td>
<td>Intel 3.1GHz processor.</td>
</tr>
<tr>
<td>Random Access Memory (RAM)</td>
<td>512 MB RAM.</td>
</tr>
<tr>
<td>Hard Disk Drive</td>
<td>20GB (min. free useable disk space). Network( NIC)</td>
</tr>
</tbody>
</table>

2) SYSTEM INTERFACES AND CONSTRAINTS

2.1) Implementation of System :-

Implementation is done in Java over Windows platform. Implementing RMI, threads and user interfaces. The Client/Server connectivity is established by creating http request and http response object. Following components are involved in system. Here the concepts of web services implemented.

1) User
2) Sponsor
3) Admin
4) Network

2.2) User Interfaces

GUI Interface

A consistent GUI will be available for all the interactions that are to be done with the system. The GUI provides an easy to understand and work interface, which assists the user to accomplish his task effectively. The GUI will be flexible enough to accommodate changes according to the user. In our case we consider the HTML, DHTML and CSS for creation of the flexible, dynamic and user friendly user interface.

2.3) Software Interfaces

The development of the system on the Java platform gives an added advantage of having the support of large number of Open Source Libraries to be used within the system. These libraries are tried and tested and will help in quick development of the system. This library of APIs will be used for the development of front end and the functionalities it provides would be useful for the purposes of building of an appreciably functional GUI.

J2SE

1) http://www.wisegeek.com/what-is-j2se.htm
2) http://www.apl.jhu.edu/~hall/java/Servlet-Tutorial/

2.4) Communications Interfaces

With the networking devices in place, the next thing to check is that how the server and client machines will communicate; for that one of the following two protocols is needed.

1. Transmission Control Protocol / Internet Protocol
2. User Datagram Packet

Both these protocols can be used for communicating. The difference between them is that TCP/IP is a connection oriented service whereas UDP is connectionless service.

The product also uses the other existing protocols and services such as JRMP.

2.5) Memory Constraints

The application being developed in Java has the following memory constraint: Java application tends to run slightly slow because they run inside the JVM. This lag can be reduced by efficient coding and avoiding use of heavy components.

Desktop Application
Minimum – 256 MB
Recommended – 512 MB Memory
IV. SYSTEM DESIGN

1) System Architecture

3) Class Diagram:

In the Unified Modeling Language (UML), a class diagram is a special type of static structure diagram. Actually this describes the exact structure of a system by showing the system’s classes, their attributes, and the relationships between the classes. So the following diagram shows the exactly the same thing about the conference system. Thus the UML class diagrams show the classes of the system, their inter-relationships, and the operations and attributes of the classes.

4) Object Diagram:-

Basically the object diagrams model instances of classes which are mainly used to describe the system at a particular point in time. Also these diagrams are closely related to the class diagram only with difference that the later one shows the objects of the classes in the class diagram. These diagrams also called as the instance diagrams sometimes.
4) Use case diagram:

This is another type of the UML diagram. A use case diagram is a type of behavioral diagram which is defined by the Unified Modeling Language (UML) and created from a Use-case analysis methodology. The main purpose of this diagram is to present a graphical overview of the functionality that one is provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. Thus the use case diagram is made up of following core components:

5) Use cases,

6) Actors,

7) Associations,

8) System Boundary boxes (optional)

9) Packages (optional)

In the following section we will see the use case diagram for the conference site with the users and the admin are the actors:

Use case diagram
5) **Activity Diagram:**

We may consider that activity diagrams are the object-oriented equivalent of flow charts and data-flow diagrams (DFDs). They are used to explore the logic of:

- A complex operation
- A complex business rule
- A single use case
- Several use cases
- A business process
- Software processes

So in general, an activity diagram is a diagram that shows activities and actions to describe workflows. In the Unified Modeling Language the concept of activity diagram used to represents the business and operational step-by-step workflows of components in a system. In short, the activity diagram shows the overall flow of system.

For our system below are the two activity diagrams that shows the conference system basic flow.
Activity diagram

USER/SPONSOR    ADMIN

Register Request → Register User → Modify Status

Login → Authenticate

Do transactions → Alerts/messages

Logout

6) Flow diagram

ADMIN

LOGIN

Do Transaction

Confirm

Normal Processing

Update Database

Not confirm

Wait for the topic confirmation

if status = y
7) Deployment Diagram:

A UML deployment diagram is the static behavioral diagram which shows a static view of the run-time configuration of hardware nodes and the software components that run on those nodes. Following is the deployment diagram for the conference system.

V. SCREENSHOTS

Home Page:
VI. TESTING

Types of Testing:

Along with the type of testing also mention the approach to be followed for the testing, i.e. Manual Testing or Automated Testing. Use Automated Testing Plan for planning automation activities in details. The different types of testing that may be carried out in the project are as follows:

1. Smoke Testing
2. Unit Testing
3. Integration Testing
4. Validation Testing
5. System Testing
6. Regression Testing
7. Load Testing
8. Stress Testing
9. Alpha testing
10. White Box Testing
11. Black Box Testing
12. GUI Testing

The different types of testing that may be carried out in the project are as follows:

✓ **Smoke testing:**
   Smoke test is an initial set of tests that determine if a new software build is performing well enough to accept it for a major testing effort. It verifies the major functionality at high level. The Smoke test scenarios emphasize more on breadth than the depth. If the test fails, the build is returned to developers without testing.

✓ **Unit testing**
   Unit testing focuses verification effort on the smallest unit of software design – the software component or module. The relative complexity of tests and uncovered errors is limited by the constrained scope established for unit testing. The unit test is white-box oriented, and the step can be conducted in parallel for multiple components.

✓ **Integration testing**
   Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. In incremental integration the program is constructed and tested in small increments, where errors are easier to isolate and correct; interfaces are more likely to be tested.

✓ **Validation testing**
   Validation can be defined in many ways, but a simple definition is that succeeds when software functions in a manner that can be reasonably expected by the customer.
   Software validation is achieved through a series of black-box tests that demonstrate conformity with requirements. A test plan outlines the classes of tests to be conducted and a test procedure defines specific test cases that will be used to demonstrate conformity with requirements. Both the plan and procedure are designed to ensure that all functional requirements are satisfied, all behavioral characteristics are achieved, all performance requirements are attained, documentation is correct, and human engineered and other requirements are met.

✓ **System Testing:**
   It tests the system as a whole. It is a functional testing, performed to validate that the application meets requirement specifications.

✓ **Regression Testing:**
   It is re-testing of a program after doing the modification. It helps in ensuring that faults have not been introduced or uncovered as a result of the changes made, and that the modified system still meets its requirements. It is performed whenever the software or its environment is changed.

✓ **Load Testing**
   It involves testing an application under heavy loads so that you can predict its working under different user loads. It also determines the point at which the system response time degrades or fails.

✓ **Stress Testing**
   It is a testing conducted to evaluate a system or component at or beyond the limits of its specified requirements (for example, extreme processor utilization, insufficient memory, inadequate hardware, dependency on over-utilized shared resources).

✓ **Alpha testing**
   The alpha test is conducted at the developer's site by a customer. The software is used in natural setting with the developer “looking over the shoulder” of the user and recording errors and usage problems. Alpha tests are conducted in a controlled environment.

✓ **White box testing**
   White-box test design allows one to peek inside the "box", and it focuses specifically on using internal knowledge of the software to guide the selection of test data. Synonyms for white-box include: structural, glass-box and clear-box. White box testing is much more expensive than black box testing. It requires the source code to be produced before the tests can be planned and is much more laborious in the determination of suitable input data and the determination if the software is or is not correct. This testing is concerned only with testing the software product; it cannot guarantee that the complete specification has been implemented.

✓ **Black box testing**
   Black-box test design treats the system as a "black-box", so it doesn't explicitly use knowledge of the internal structure. Black-box test design is usually described as focusing on testing functional requirements. Synonyms for black box include: behavioral, functional, opaque-box, and closed-box.
   Black box testing is concerned only with testing the specification; it cannot guarantee that all parts of the implementation have been tested. Thus black box testing is testing against the specification and will discover faults of omission, indicating that part of the specification has not been fulfilled.

✓ **GUI testing**
   GUI (Graphical User Interface) testing are the testing guidelines for specialized environments, architectures and application data commonly encountered by software engineers. Because of reusable components provided as a part of GUI
development environments, the creation of user interface has become less time consuming and more precise. But the complexity of GUI's has also grown, leading to more difficulty in the design and execution of test cases. As modern GUIs have the same look and feel, a series of standard tests can be derived. The formal technical review, so conducted was an umbrella activity and served as a part of the Software Quality Assurance Plan.

8.6 Test plan
Modules are to be built in an incremental manner. Each module is then to be tested individually (unit testing). Testing is to be done to ensure that the module meets its required specifications and performs up to the desired level. Then modules are to be integrated. During integration itself, integration testing is to be carried out on the individual modules. After integration, regression testing is to be performed to ensure that the integration does not introduce new bugs. Finally when all modules are put in place, alpha testing is to be carried out on the entire system. Stress testing is also required to be done to ensure that the system lives up to its desired performance and conforms to the requirements. This plan was followed rigorously and bugs in the code were traced and eliminated. The test cases used and the results obtained at the end of all testing activity are given below in a tabular format for ease in comprehension.

8.6.1 Features to be tested
1. Check for input files selection. Validate whether it functions as per the requirement.
2. All select-deselect buttons works efficiently.
3. User can efficiently interact with system.
4. All modules functionally correct.
5. Handle all types of user interactions.
6. The user interaction should be reflected to the database.
7. Checks the extensibility features provided and confirm whether it works correctly.
8. Check that system works efficiently even in case of multiple selections of parameters.
9. Check for report generation functionality.

VII. CONCLUSION
E-Campus System allows not only a job seeker to get latest and correct placement information, but it also helps job recruiters to get the correct list of candidates according to their criteria posted through java messaging services using the J2ME based technologies. the online campus program which will be used by the many colleges, placement agencies and many more with advance messaging and alerts to the both recruiters and the job seekers about the latest information.

VIII. SUMMARY
Software project management is an umbrella activity. It begins with technical basics activity. Requirement analysis is the first step in the software process. It is at this point that a general statement of software scope is refined into concrete specifications. The Software Requirements Specification is developed as a consequence of analysis. Project planning begins with how long project will take how much efforts will be required and predict the resources (hardware and software) that will be required and identifying risks involved. Risk analysis can absorb a significant amount of project planning. Design is the technical kernel of software engineering. Design document contains various UML diagrams. Software testing accounts for the largest percentage of technical efforts in the software process. The objective of software testing is to uncover errors. To fulfill this objective, a series of test steps – Unit, integration, validation, system tests, etc. are planned and executed. Then the screen shots of project are shown giving the idea of, how project works.

To implement project
The JSP is used for front end design. Microsoft SQL Server 2000 as backend. Sending and receiving messages are tested successfully. Status reports are generated successfully.

IX. REFERENCES
[2]. Thinking in Java by Bruce Eckel.
[3]. Java servlet books: A comparative review Find out where you can get the best servlet coverage
[4]. By John Zukowski, JavaWorld.com, 03/08/00.
[10]. Professional JSP by Wrox publication.
[16]. Building Dynamic HTML GUIs (Paperback) by Steven Champeon (Author), David S. Fox (Author).
[17]. Creating Cool HTML 4 Web Pages (Paperback) by Dave Taylor (Author).
Appendix:

Index:
1)  INTRODUCTION
2)  SYSTEM PHASES
3)  TECHNOLOGY DESCRIPTION
4)  SOFTWARE REQUIREMENT SPECIFICATIONS
5)  SYSTEM DESIGN
5.1 System Architecture
5.2 Class Diagram
5.3 Object Diagram
5.4 Use Case Diagrams
5.5 Activity Diagram
5.6 Deployment Diagram
6)  SCREENSHOTS
7)  TESTING
8)  Demo Source Code
9)  CONCLUSION
10) SUMMARY
11) REFERENCE