Hi –Fi Infirmary Portal Java Project Report

G.Vidhya Ratna Sagar
Student(M.Tech) , CSE,
Gokul Institue of Technology and Science,
Visakhapatnam, India.

A.Achutharao
Asst. Prof, CSE,
Gokul Institue of Technology and Science,
Visakhapatnam, India.

ABSTRACT:

In this project an efficiency of any system designed to suit an organization depends cooperation during the implementation stage and also flexibility of the system to adopt itself to the organization is provided. High-fi Infirmary Software takes care of the activities related to the schemes provided by the government.

The major responsibility is to take care of the proper utilization of different schemes in a transparent way. Reduction in the time spent by the staff to gather the information & to store it in an organized manner.Also reduction in the time for generating reports related to various activity like diagnosis report, death certificates, birth certificates, fitness certificates etc.,

I INTRODUCTION:

Its main purpose it to maintain information about hospitals and patients,doctors,etc.In present situation lot of hospitals maintains their personal data either by using manual records or by using computers. But they are not providing any online interface to all patients to view their personal details like test reports, deceases etc. To avoid this situation and to provide online communication between busy people to hospital management we are introducing a new portal called "High-fi Infirmary".

1.1 SCOPE:

This document shows the requirement and functional document for High-fi Infirmary. It symbolizes all the needs and the requirements for the client [Global Life Clinic] to achieve a quality that is desired for the application and it contains information for the development team to design the application.

1.2 SYSTEM ARCHITECTURE:
Architecture flow:

Below architecture diagram represents mainly flow of requests from users to database through servers. In this scenario overall system is designed in three tires separately using three layers called presentation layer, business logic layer and data link layer.

This project was developed using 3-tier architecture.

1.3 TECHNICAL FEASIBILITY:

The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipments have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?

Are there technical guarantees of accuracy, reliability, ease of access and data security?

1.4 OPERATIONAL FEASIBILITY:

Customer will use the forms for their various transactions i.e. for adding new routes, viewing the routes details. Also the Customer wants the reports to view the various transactions based on the constraints. These forms and reports are generated as user-friendly to the Client.

The package wills pick-up current transactions on line. Regarding the old transactions, User will enter them in to the system.

The application will be developed using standard open source software (Except Oracle) like Java, tomcat web server, Internet Explorer Browser etc these software will work both on Windows and Linux o/s. Hence portability problems will not arise.

The system called the ewheelz uses the 2-tier architecture. The 1st tier is the GUI, which is said to be front-end and the 2nd tier is the database, which uses MySql, which is the back-end. The front-end can be run on different systems (clients).

The database will be running at the server. Users access these forms by using the user-ids and the passwords.

1.5 ECONOMIC FEASIBILITY:

The computerized system takes care of the present existing system's data flow and procedures completely and should generate all the reports of the manual system besides a host of other management reports. It should be built as a web based application with separate web server and database server. This is required as the activities are spread throughout the organization customer wants a centralized database. Further some of the linked transactions take place in different locations.

Open source software like TOMCAT, JAVA, MySQL and Linux is used to minimize the cost for the Customer.

II ABOUT JAVA:

Initially the language was called as "oak" but it was renamed as "Java" in 1995. The primary motivation of this language was the need for a platform-independent (i.e., architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

Java is a programmer's language.

Java is cohesive and consistent.

Except for those constraints imposed by the Internet environment, Java gives the programmer, full control.

Finally, Java is to Internet programming where C was to system programming.

2.1 IMPORTANCE OF JAVA TO THE INTERNET:

Java has had a profound effect on the Internet. This is because; Java expands the Universe of objects that can move about freely in Cyberspace. In a network, two categories of objects are transmitted between the Server and the Personal computer.

They are: Passive information and Dynamic active programs. The Dynamic, Self-executing programs cause serious problems in the areas of Security and probability. But, Java addresses those concerns and by doing so, has opened the door to an exciting new form of program called the Applet.

2.2 JAVA CAN BE USED TO CREATE TWO TYPES OF PROGRAMS:

APPLICATIONS AND APPLETS: An application is a program that runs on our Computer under the operating system of that computer. It is more or less like one creating using C or C++. Java's ability to create Applets makes it important. An Applet is an application designed to be transmitted over the Internet and executed by a Java –compatible web browser. An applet is actually a tiny Java program, dynamically downloaded across the network, just like an image. But the difference is, it is an intelligent program, not just a media file. It can react to the user input and dynamically change.

2.3 FEATURES OF JAVA SECURITY:

Every time you that you download a "normal" program; you are risking a viral infection. Prior to Java, most users did not download executable programs frequently, and those who did scan them for viruses prior to execution. Most users still worried about the possibility of infecting their systems with a virus. In addition, another type of malicious program exists that must be guarded against. This type of program can gather private information, such as credit card numbers, bank account balances, and passwords. Java answers both these concerns by providing a "firewall" between a network application and your computer..

PORTABILITY:

For programs to be dynamically downloaded to all the various types of platforms connected to the Internet, some means of generating portable executable code is needed .As you will see, the same mechanism that helps ensure security also helps create portability. Indeed, Java's solution to these two problems is both elegant and efficient.

THE BYTE CODE:

The key that allows the Java to solve the security and portability problems is that the output of Java compiler is Byte code. Byte code is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for byte code.

JAVA VIRTUAL MACHINE(JVM):

Beyond the language, there is the Java virtual

machine. The Java virtual machine is an important element of the Java technology. The virtual machine can be embedded within a web browser or an operating system. Once a piece of Java code is loaded onto a machine, it is verified. As part of the loading process, a class loader is invoked and does byte code verification makes sure that the code that's has been generated by the compiler will not corrupt the machine that it's loaded on. Byte code verification takes place at the end of the compilation process to make sure that is all accurate and correct. So byte code verification is integral to the compiling and executing of Java code.

Overall Description:

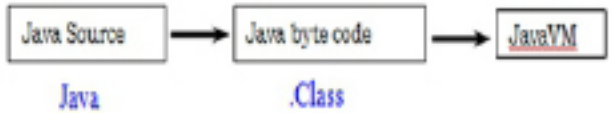

Fig 1 Picture showing the development process of JAVA Program.

Java programming uses to produce byte codes and executes them. The first box indicates that the Java source code is located in a. Java file that is processed with a Java compiler called javac. The Java compiler produces a file called a. class file, which contains the byte code. The. Class file is then loaded across the network or loaded locally on your machine into the execution environment is the Java virtual machine, which interprets and executes the byte code.

III JAVA ARCHITECTURE:

Java architecture provides a portable, robust, high performing environment for development. Java provides portability by compiling the byte codes for the Java Virtual Machine, which is then interpreted on each platform by the run-time environment. Java is a dynamic system, able to load code when needed from a machine in the same room or across the planet.

3.1 COMPILATION OF CODE:

When you compile the code, the Java compiler creates machine code (called byte code) for a hypothetical machine called Java Virtual Machine (JVM). The JVM is supposed to execute the byte code. The JVM is created for overcoming the issue of portability. The code is written and compiled for one machine and interpreted on all machines. This machine is called Java Virtual Machine.

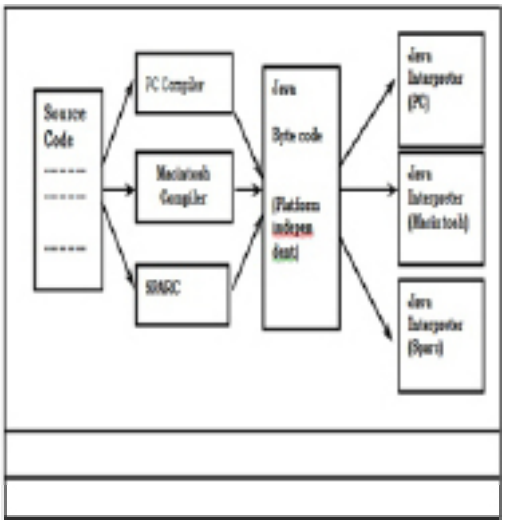

Fig 2 compilation of code.

During run-time the Java interpreter tricks the byte-code file into thinking that it is running on a Java Virtual Machine. In reality this could be a Intel Pentium Windows 95 or SunSARC station running Solaris or Apple Macintosh running system and all could receive code from Java virtually eliminates the problems of memory management and deallocation, which is completely automatic. In a well-written Java program, all run time errors can –and should –be managed by your program.

3.2 SERVLETS:

The Java web server is JavaSoft's own web Server. The Java web server is just a part of a larger framework, intended to provide you not just with a web server, but also with tools. To build customized network servers for any Internet or Intranet client/server system. Servlets are to a web server, how applets are to the browser.

3.3 ABOUT SERVLETS:

Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming, including inextensible scripting solutions, platform-specific APIs, and incomplete interfaces. Servlets are objects that conform to a specific interface that can be plugged into a Java-based server. Servlets are to the server-side what applets are to the client-side - object byte codes that can be dynamically loaded off the net.

They differ from applets in that they are faceless objects (without graphics or a GUI component). They serve as platform independent, dynamically loadable, plugable helper byte code objects on the server side that can be used to dynamically extend server-side functionality.

IV LOADING SERVLETS:

Servlets can be loaded from three places From a directory that is on the CLASSPATH. The CLASSPATH of the JavaWebServer includes service root/classes/ which is where the system classes reside.

From the <SERVICE_ROOT /Servlets/ directory. This is *not* in the server's classpath. A class loader is used to create Servlets from this directory. New Servlets can be added - existing Servlets can be recompiled and the server will notice these changes.

From a remote location. For this a code base like http: // nine.eng / classes / foo / is required in addition to the Servlets class name. Refer to the admin GUI docs on Servlet section to see how to set this up.

4.1 LOADING REMOTE SERVLETS:

Remote Servlets can be loaded by:

1.Configuring the Admin Tool to setup automatic loading of remote Servlets .

2.Setting up server side include tags in .shtml files .

3.Defining a filter chain configuration .

4. INVOKING SERVLETS:

A Servlet invoker is a Servlet that invokes the "service" method on a named Servlet. If the Servlet is not loaded in the server, then the invoker first loads the Servlet (either from local disk or from the network) and the then invokes the "service" method. Also like applets, local Servlets in the server can be identified by just the class name. In other words, if a Servlet name is not absolute, it is treated as local.

V TWO-TIER AND THREE-TIER MODELS:

The JDBC API supports both two-tier and three-tier models for database access. In the two-tier model, a Java applet or application talks directly to the database. This requires a JDBC driver that can communicate with the particular database management system being accessed. A user's SQL statements are delivered to the database, and the results of those statements are sent back to the user. The database may be located on another machine to which the user is connected via a network. This is referred to as a client/server configuration, with the user's machine as the client, and the machine housing the database as the server.

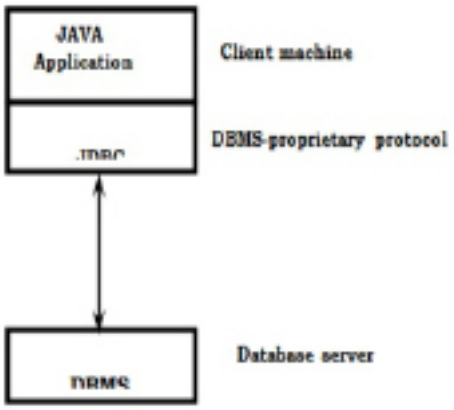

Fig 3 Middle tier architecture.

Until now the middle tier has typically been written in languages such as C or C++, which offer fast performance. However, with the introduction of optimizing compilers that translate Java byte code into efficient machine-specific code, it is becoming practical to implement the middle tier in Java.

This is a big plus, making it possible to take advantage of Java's robustness, multithreading, and security features. JDBC is important to allow database access from a Java middle tier.

5.1 SYSTEM TESTING:

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During software development. During testing, the program is executed with a set of test cases and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected to perform.

5.2 TESTING IN STRATEGIES:

In order to make sure that the system does not have errors, the different levels of testing strategies that are applied at differing phases of software development are:

Unit Testing:

Unit Testing is done on individual modules as they are completed and become executable. It is confined only to the designer's requirements.

Each module can be tested using the following two Strategies:

5.3 Black Box Testing:

In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been uses to find errors in the following categories:

- Incorrect or missing functions.
- Interface errors .
- Errors in data structure or external database access.
- Performance errors.
- Initialization and termination errors.

5.4 White Box testing:

In this the test cases are generated on the logic of each module by drawing flow graphs of that module and logical decisions are tested on all the cases. It has been uses to generate the test cases in the following cases:.

5.5 Integrating Testing:

Integration testing ensures that software and subsystems work together a whole. It tests the interface of all the modules to make sure that the modules behave properly when integrated together.

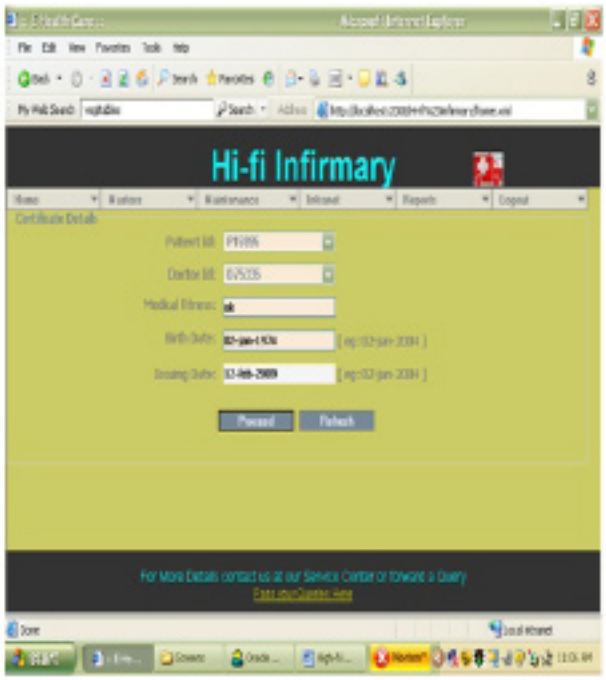

Fig 4 Screenshot for Certificate Details

This screenshot provides different varieties of certificates generally issued by the diagnostics like fitness, birth, death certificates etc., by showing these certificates doctor finalize that what problem they have & what to do next.This screenshot is used to maintain the schedules of the doctor to words patient, and appointments, this will be maintained by hospital administrator only which helps to avoid confusion from doctor schedules.

VI CONCLUSION:

The efficiency of any system designed to suit an organization depends cooperation during the implementation stage and also flexibility of the system to adopt itself to the organization. High-fi Infirmary Software takes care of the activities related to the schemes provided by the government. The major responsibility is to take care of the proper utilization of different schemes in a transparent way. Reduction in the time spent by the staff to gather the information & to store it in an organized manner.Also reduction in the time for generating reports related to various activity like diagnosis report, death certificates, birth certificates, fitness certificates etc.,

VII REFERENCES:

JAVA Technologies.

JAVA Complete Reference.

Java Script Programming by Yehuda Shiran.

Mastering JAVA Security.

JAVA2 Networking by Pistoria.

JAVA Security by Scotl oaks.

Head First EJB Sierra Bates.

J2EE Professional by Shadab siddiqui.

JAVA server pages by Larne Pekowsley.

JAVA Server pages by Nick Todd.

HTML.

HTML Black Book by Holzner.

JDBC.

Java Database Programming with JDBC by Patel moss.

Software Engineering by Roger Pressman.