**CHAPTER 1**

**INTRODUCTION**

Webcams are popular, relatively low cost devices which can provide live video and audio streams via personal computers, and can be used with many software clients for both video calls and videoconferencing. LAN Videoconferencing uses audio and video telecommunications to bring people at different sites together which are connected through local area network. This can be as simple as a conversation between people in private offices (point-to-point) or involve several (multipoint) sites in large rooms at multiple locations. Besides the audio and visual transmission of meeting activities text chat facility is also present.

High speed Internet connectivity has become more widely available at a reasonable cost and the cost of video capture and display technology has decreased. Consequently, personal videoconferencing systems based on a webcam, personal computer system, software compression and broadband Internet connectivity have become affordable to the general public.

With the introduction of relatively low cost , high capacity broadband and telecommunication services in the late 1990s, coupled with powerful computing processors and video compression techniques, VideoConferencing usage has made significant inroads in business, education, medicine and media. Like all long distance communications technologies (such as phone and Internet), by reducing the need to travel to bring people together the technology also contributes to reductions in carbon emissions, thereby helping to reduce global warming.

**RATIONALE**

The aim of this project is to create a suite of programs using java that implements audio and video chat on systems connected through LAN without the need of any internet connection, additional installations, creating or handling accounts and passwords.

**PROBLEM DEFINATION**

In present scenario , it is not always possible for a person to be physically present at the required place just at the moment . It is at this moment of urgency to be at a particular place that we require softwares that allow chat over a LAN and are found to be useful. It reduces the physical movements and extends a feel of comfortability to the people by allowing them to do a video chat and text through any two systems connected over a LAN. It also helps to attend meetings if a person could not be physically present at the place of meeting through Video Conferencing. It allows to instantly meet without prior notice and join meetings without anydownloads or installations. No account is needed to enable this type of conferencing.

**Proposed Solution**

LAN Videoconferencing uses audio and video telecommunications to bring people at different sites together which are connected through local area network. This can be as simple as a conversation between people in private offices (point-to-point) or involve several (multipoint) sites in large rooms at multiple locations. Besides the audio and visual transmission of meeting activities text chat facility is also present.

* 1. **REPORT ORGANIZATION**

This project is organized as follows.

1. The introduction to the thesis is given in Chapter 1. This section describes the problem definition and proposed solution.
2. Chapter 2 reviews numerous existing and working systems. This section also shows a fair comparison between the old and traditional ways of how the live chat is completed.
3. Chapter 3 This section gives the detailed information of our requirements both hardware and software.
4. Chapter 4 draws conclusions from the work described in previous chapters and discusses possibilities for future development.
5. Chapter 5 describes the various results that are obtained after the complete implementation of the project.
6. Chapter 6 shows the testing done on our project. The strategy we have apdoted for this project testing.
7. Chapter 7 gives the screenshot of our project and the important coding used to build this project.
8. Chapter 8 shows complete conclusion and future enhancement.

Finally, in the last section i.e. Appendix, we have provided the annotated bibliography. This section is an exhaustive list of the references we have used to make this project.

**CHAPTER 2**

**FRAMEWORK**

Any data that changes meaningfully with respect to time can be characterized as time based media. Audio-clips, movie clips and animation are common form of time-based media. They can be obtained from various sources like network files, camera, microphones, & live broadcasts.

Time-based media is also referred as streaming media- it is delivered in a steady stream that must be received & processed within a particular time frame to produce acceptable results. Media data is in media streams that are obtained from a local file, acquired over network or captured from a camera or microphone.

**Advantages of Framework**

The core technology used in a videoconferencing system is digital compression of audio and

video streams in real time. The hardware or software that performs compression is called

a codec (coder/decoder). Compression rates of up to 1:500 can be achieved. The resulting digital stream of 1s and 0s is subdivided into labeled packets, which are then transmitted through a digital network of some kind (usually ISDN or IP).

VideoConferencing adds another possible alternative, and can be considered when:

* a live conversation is needed;
* non-verbal (visual) information is an important component of the conversation;
* the parties of the conversation don’t need to physically come to the same location;
* Deaf, hard-of-hearing and mute individuals have a particular interest in the development of afford ble high-quality videoconferencing as a means of communicating with each other in sign language.

Videoconferencing provides students with the opportunity to learn by participating in two-way communication forums. Furthermore, teachers and lecturers can be brought to remote or otherwise isolated educational facilities. Students could easily attend their lectures from their hostels or being anywhere in the college.

A few examples of benefits that videoconferencing can provide in campus environments include:

* faculty members keeping in touch with classes while attending conferences
* Virtual meeting room with feature-rich controls in synchronization with all video and web tools.
* guest lecturers brought in classes held in different part of the same institution
* researchers collaborating with colleagues at institutions on a regular basis without loss of time due to travel
* schools with multiple campuses collaborating and sharing professors
* faculty members participating in thesis defenses at the institution

* administrators on tight schedules collaborating on budget preparation from different parts of campus;
* faculty committee auditioning scholarship candidates;
* researchers answering questions about grant proposals from agencies or review committees;
* interviews
* teleseminars.

Videoconferencing is a highly useful technology for real- time telemedicine and telenursing applications, such as diagnosis, consulting, transmission of medical images, etc... With videoconferencing, patients may contact nurses and physicians in emergency or routine situations; physicians and other paramedical professionals can discuss cases across large distances.

Videoconferencing can enable individuals in distant locations to participate in meetings on short notice, with time and money savings. Technology such as VoIP can be used in conjunction with desktop videoconferencing to enable low-cost face-to-face business meetings without leaving the desk, especially for businesses with widespread offices.

**2.1 BACKGROUND**

1. MEAN Stack(Mongo, Express, Angular, Node).
2. Sockets to enable one-on-one communication in real time
3. AJAX for sign-up and login

**2.2 RELATED WORK**

Following are the functionalities of the LAN CHAT COMMUNICATOR:-

* LOGIN :- User have their separate login id and password.
* REGISTER:- Users can register for chat services
* PERSONAL CHAT:- Users can chat at personal level.
* GROUP CHAT :- Users can also chat in group.

**CHAPTER 3**

**SYSTEM ANALYSIS**

This section describes about pre-requisite for systems of our project also the feasibility of our system and development method of our project. System analysis patterns or analysis are conceptual models, leading to specifications of a new system. Analysis is a detailed study of various operations performed by a system and their relationships within and outside the system. During analysis, data are collected on the available files, decision points and transactions handled by the present system.

**3.1 HARDWARE REQUIREMENTS**

Processor : Pentium,AMD or Higher Version

Operating System : Windows XP/ Windows 7/ Linux

RAM : 256 MB, 2GB recommended

Hardware Devices : Keyboard with mouse

Hard disk : 10 GB or more

Display : Standard Output Display

* 1. **SOFTWARE REQUIREMENTS**

Technology Implemented MEAN Stack(Mongo, Express, Angular, Node).

Sockets to enable one-on-one communication in real time

AJAX for sign-up and login

Database : MongoDb

User Interface Design : HTML5,CSS, JavaScript

Web Browser : Mozilla, Google Chrome, Internet explorer

**3.3 SERVER SIDE**

Processor : Pentium, AMD or Higher Version

Speed : 1.0 GHz

Memory : 256 MB

Hard Disk : 40GB Hard disk with minimum 4GB free space

**3.4 CLIENT SIDE**

Operating System : Windows XP/ Windows 7/ Linux

Application type : Web Application

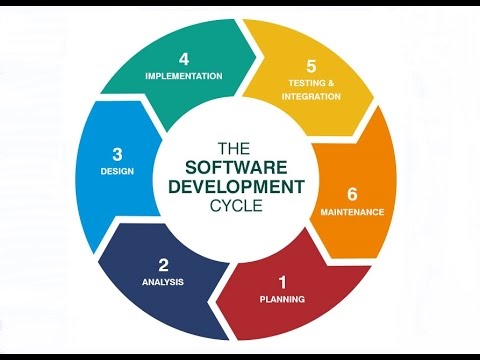
**CHAPTER 4**

**SYSTEM ANALYSIS AND DESIGN**

Based on the user requirements and the detailed analysis of a system, the system is designed. This is the phase of system designing. It is a crucial phase in the development of a system. It includes the UML diagrams.

**4.1 SYSTEM ANALYSIS**

System development has two major component- system analysis and system design. It refers to the process of examining a business situation with the intent of improving it methods and procedure.

****

**4.1.1 REQUIREMENT ANALYSIS**

Requirements analysis in systems engineering and software engineering, encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, analyzing, documenting, validating and managing software or system requirements. Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

**4.1.2 OBJECT ORIENTED ANALYSIS**

Object Oriented Analysis is the first technical activity that is performed as a part of object oriented software engineering. Object Oriented Analysis introduces new concepts for examining a problem. It is grounded in a set of basic principles, which are as follows

* The information domain is modeled
* Behavior is represented.
* Function is described.
* Data, functional and behavioral models are divided to uncover greater detail
* Early models represent the essence of the problem, while later ones provide implementation details.

**4.1.3 ARCHITECTURAL SPECIFICATION**

A description of a software and electronics system in terms of its hardware and software components and their interactions. In (hardware, software, or enterprise) systems development, an architectural specification is the set of documentation that describes the structure, behavior, and more views of that system. The architectural specification include the report organization of the project. The role of the system architectural structure that serve as the blueprint for the target system.

* Allocate required system functionality to hardware and software.
* Document crucial design constraints, assumptions and rationales.
* Support early analysis to make sure that the design approach is highly visible and open to peer review and progressive improvement.
* Demonstrate compliance with system requirements. The SAAS provides an authoritative reference for detailed traceability analysis.
* Explain how the end product will exhibit required qualities such as usability, performance, modifiability, safety and security.
* Describe design management strategies to be used to control the development of the design including nomination of design patterns and rules.
* Support project planning and budgeting.
* Support preparation of acquisition documents (for example, requests for proposal and¬ statements of work)
* Support on-going maintenance and enhancement.

**4.1.4 FEASIBILITY ANALYSIS**

Feasibility analysis (FA, also called feasibility study) is used to assess the strengths and weaknesses of a proposed project and present directions of activities which will improve a project and achieve desired results. The nature and components of feasibility studies depend primarily on the areas in which analyzed projects are implemented. The evaluation of legal requirements concerns basically the rules of publishing and using data, software and methods. A part of the rules concerning publishing spatial data (mostly future) can be found in the INSPIRE directive and in copyright laws. Most data and software include information on the form of publishing in metadata or websites where such data is published. These components are important in the evaluation of economic requirements:

* The cost of data or software procurement.
* The cost of employing workforce for specific tasks.
* The cost of possible improvement of qualifications of the workforce employed.

The evaluation of requirements related to the schedule concerns the estimation of time necessary to complete respective parts of a project, e.g. data procurement. Feasibility analysis is achieved by checking the load over the application i.e. if 20-25 no. of user login at the same time the system does not crash. Economic requirement include:

* No hardware used thus hardware cost is reduced.
* Small app does not require any workforce.
* No highly professionals are required. Basic knowledge of java and mysql is sufficient to¬ built the website.

**4.1.5 DEVELOPMENT METHOD**

After designing the new system, the whole system is required to be converted into computer understanding language. Coding the new system into computer programming language does this. It is an important stage where the defined procedures are transformed into control specifications by the help of a computer language. This is also called the programming phase in which the programmer converts the program specifications into computer instructions, which refer as programs. The programs coordinate the data movements and control the entire process in a system.

My development occurred in following phases

* Platform selection: HTML, CSS, Java Script, MongoDb.
* Observed a tutorial site of w3schools.com.
* Created drawings showing the future system.
* Created new design of the system .

**4.2 SYSTEM DESIGN**

The System Design Document describes the system requirements, operating environment, system and subsystem architecture, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces

**4.2.1 USE CASE DIAGRAM**

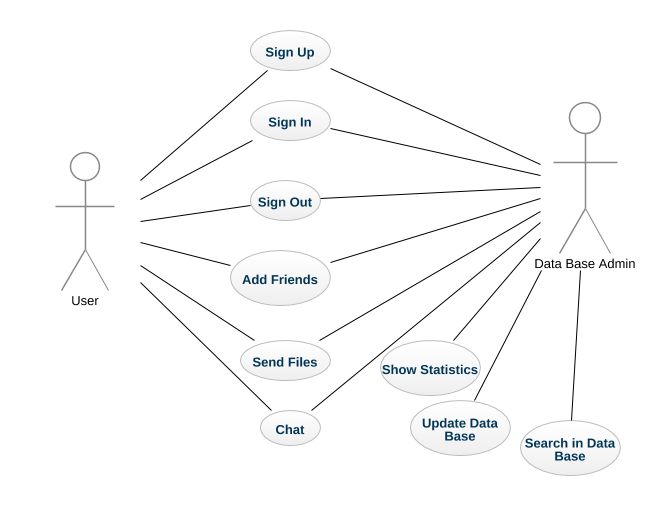
A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted. Use Case diagrams are formally included in two modeling languages defined by the OMG: the Unified Modeling Language (UML) and the Systems.

**4.2.2 USE CASE SPECIFICATION**

* Application: LAN CHAT COMMUNICATOR
* Use-case name: User
* Use-case description: This use case details the category for selecting a service where the user can use the services of personal chat and group chat.
* Primary actor: User
* Precondition: The user successfully runs the application for the chat purpose
* Post-condition: The user successfully selects/deselects a service for completion of his need.

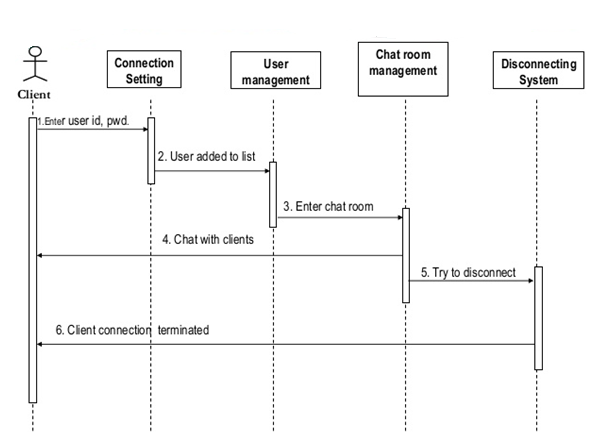
**Basic Flow:**

* Run the application
* Login
* Create the connection
* Select the service of personal chat or group chat
* Chat
* Disconnect



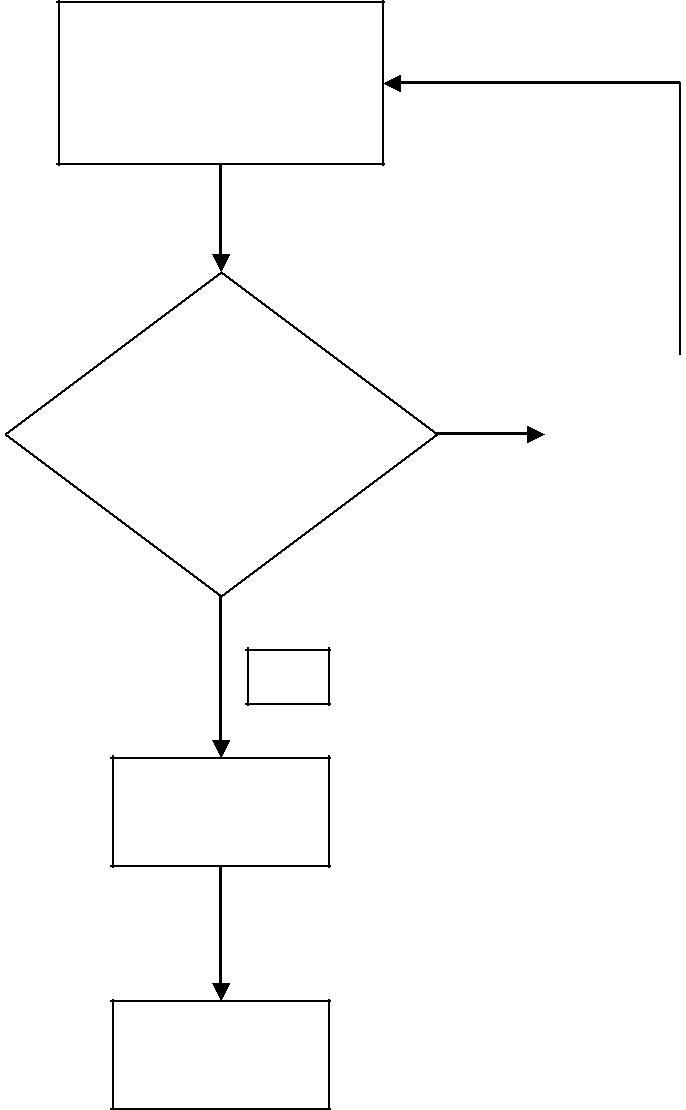
**4.2.3 SEQUENCE DIAGRAM**

A sequence diagram is a graphical view of a scenario that shows object interaction in a timebased sequence, what happens first, what happens next. Sequence diagrams establish the roles of objects and help provide essential information to determine class responsibilities and interface. This type of diagram is best used during early analysis phases because they are simple and easy to comprehend. A sequence diagram has two dimensions: typically, vertical placement represents time and horizontal placement represents different objects.

****

**4.2.4 ACTIVITY DIAGRAM**

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system. Activity diagrams provide a way to model the workflow of a business process. Activity diagrams can also be used to model code-specific information, such as a class operation. Activity diagrams are very similar to a flowchart because of modelling a workflow from activity to activity. An activity diagram is basically a special case of a state machine in which most of the states are activities and most of the transitions are implicitly triggered by completion of the actions in the source activities. Each activity represents the performance of a group of actions in a workflow. Once the activity is complete, the flow of control moves to the next activity or state through a transition.



Client enters login name & password

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Login Name |  | **Y** |  | Report Client |
| Exists? |  |  |  | about Existence |
|  |  |  |  |  |

**N**

Enter into

Array

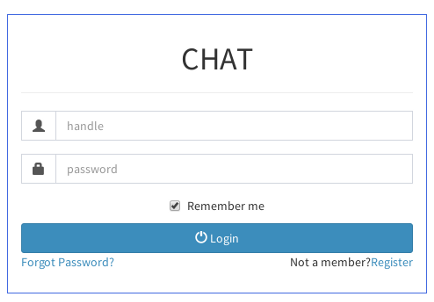
Enter into

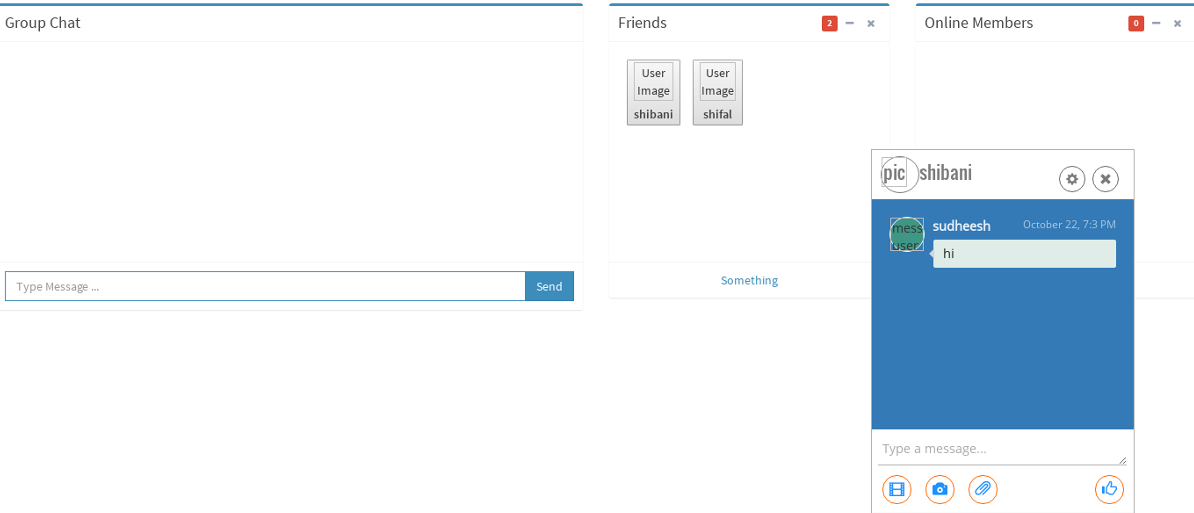
Database

**CHAPTER 5**

**PROJECT IMPLEMENTATION AND OUTPUT SCREEN**

* 1. **SCREENSHOTS**





* 1. **IMPORTANT CODING**

**CHAPTER 6**

**TESTING**

* 1. **Testing**

The goal of this document is to develop a test design for the Classified Website an web application. This document defines all the procedures and activities required to prepare for testing of the functionalities of the system which are specified in the document. The objectives of the test plan are to define the activities to perform testing, define the test deliverables documents and to identify the various risks and contingencies involved in testing. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation. Once source code has been generated software must be tested to uncover and correct as many errors as possible before delivery to the customer. Our goal is to design a series of test case that have a high likelihood of finding errors. That’s where software testing techniques come into the picture. These techniques provided systematic guidance for designing test that

1. Exercise the internal logics of the software components.

2. Exercise the input and output domain of the program to uncover errors in program function, behavior performance.

Testing of the software leads to the uncovering of errors in the software functional and performance requirements are met. Testing also provide a good indication of software reliability as software quality as whole. The results of the different faces of testing are evaluated and then compare with the expected results. If the errors are uncovered they are debugged and corrected.

**6.2 Types of testing**

**System Testing**

A system normally consists of all component that makeup the total system to function. It will be required to ensure the smooth running of the system as a whole, and it should perform as expected and as required. Here, technical and functional testing will be performed. The technical testing will be involving the process of testing the system compatibility with the hardware, operating system, data integrity in the database and user authentication access rights.

**Unit Testing**

Unit test is where the system is tested partially and independently, component by component, to ensure that particular portion or module is workable within it.

In the development of the Wedding planning, each component will be tested independently before finally integrating each of them into one system.

**6.3 Test approaches and test plan**

Large system usually tested using a mixture of strategies. Different strategies may be needed for different parts of the system or at a stage of the process.

**Top-****down testing:** This approach tests high levels of system before detailed components. This is an appropriate when developing the system top-down likely to show up structural design errors early (and therefore cheaply) has advantage that a limited, working system available early on. Validation (as distinct from verification) can begin early. Its disadvantage is that stubs needs to be generated (extra effort) and might be impracticable if component is complex (e.g. converting an array into a linked list; unrealistic to generate random list; therefore, end up implementing unit anyway). Test output may be difficult to observe (needs creation of artificial environment). This is not appropriate for OO systems (except within a class).

**Bottom-up testing:**This is opposite of top-down testing. This testing test low-level unit then works up hierarchy. Its advantages and disadvantages of bottom-up mirror those of top-down. In this testing there is need to write test drivers for each unit. These are as reusable as the unit itself. Combining top-down development with bottom-up testing means that all parts of system must be implemented before testing can begin, therefore does not accord with incremental approach discussed above. Bottom-up testing less likely to reveal architectural faults early on. However, bottom-up testing of critical low-level components is almost always necessary. Appropriate for OO systems.

**Back-to-back testing:**Comparison of test results from different versions of the system (e.g. compare with prototype, previous version or different configuration). Process - Run first system, saving test case results. Run second system, also saving its results. Compare results files. Note that no differences don’t imply no bugs. Both systems may have made the same mistake.

**Defect testing:**  A successful defect test is a test that causes the system to behave incorrectly. Defect testing is not intended to show that a program meets its specification. If tests don't show up defects it may mean that the tests are not exhaustive enough.

Exhaustive testing is not always practicable. Subset has to be defined (this should be part of the test plan, not left to the individual programmer). Possible methods:

* Test capabilities rather than components (e.g. concentrate on tests for data loss over ones for screen layout).
* Test old in preference to new (users less effected by failure of new capabilities).
* Test typical cases rather than boundary ones (ensure normal operation works properly).

Three approaches to defect testing. Each is most appropriate to different types of component. Studies show that black box testing is more effective in discovering faults than white-box testing. However, the rate of fault detection (faults detected per unit time) was similar for each approach. Also showed that static code reviewing was more effective and less expensive than defect testing.

**6.4 Testing tools and environment**

In simple terms – **Test Environment** is nothing but a replica of actual production environment (being used by end-users) with close-enough hardware and software configurations, where the testing would happen for the developed application. **Test environment** is where all the action happens – experiments, defect identification, fixing, retesting, regression and final sign-off. Ideally. **Test environment** configuration must mimic the production environment in order to uncover any environment/configuration related issues.

Following are the testing tools

### Selenium

Selenium is a testing framework to perform web application testing across various browsers and platforms like Windows, Mac, and Linux. Selenium helps the testers to write tests in various programming languages like Java, PHP, C#, Python, Groovy, Ruby, and Perl. It offers record and playback features to write tests without learning Selenium IDE.

Selenium proudly supports some of the largest, yet well-known browser vendors who make sure they have Selenium as a native part of their browser. Selenium is undoubtedly the base for most of the other software testing tools in general.

### Ranorex

Ranorex Studio offers various testing automation tools that cover testing all desktop, web, and mobile applications.

Ranorex offers following features:

* GUI recognition
* Reusable test codes
* Bug detection
* Integration with various tools
* Record and playback

**6.5Test Cases**

Table 6.1: User Registration Form

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Registration Form** |
| Description | **This scenario covers the functionality of the registration form for new admin** |
| Module Name | **Registration** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **19:14 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Admin\_Reg | Admin Registration | Admin must have input all the valid information. | Successful registration. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.2: Admin Login Form

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Login Form** |
| Description | **This scenario covers the functionality of the login form for registered admin.** |
| Module Name | **Admin Login** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **20:12 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Admin\_Login | Admin Login | Admin must have input all the valid information. | Successful login. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.3: Add Candidate

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Add Candidate** |
| Description | **This scenario covers the functionality of adding candidates for election** |
| Module Name | **Add Candidate** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **08:11 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Add\_Candidate | Add Candidate | Admin must have submit all the valid information. | Candidate will be added to the selected category. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table6.4: Add Voter

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Add Voter** |
| Description | **This scenario covers the functionality of adding voter list for election** |
| Module Name | **Add voter** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **09:23 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Add\_Candidate | Add Candidate | Admin must have submit all the valid information. | Candidate will be added to the selected category. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.5: Add Election

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Add Election** |
| Description | **This scenario covers the functionality of adding election** |
| Module Name | **Add election** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **12:30 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Add\_Election | Add election | Admin must have submit all the valid information. | Election will be added to the selected category. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.6: Delete Candidate

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Delete Candidate** |
| Description | **This scenario covers the functionality of Deleting candidate** |
| Module Name | **Delete Candidate** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **09:45 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Delete\_candidate | Delete Candidate | Admin must have submit all the valid information. | Candidate will be deleted to the selected category. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.7: Delete voter

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Delete Voter** |
| Description | **This scenario covers the functionality of Deleting voter** |
| Module Name | **Delete voter** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **09:50 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Delete\_voter | Delete voter | Admin must have submit all the valid information. | Voter will be deleted to the selected category. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.8: Delete Candidate

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Delete election** |
| Description | **This scenario covers the functionality of Deleting candidate** |
| Module Name | **Delete election** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **10:10 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Delete\_election | Delete election | Admin must have submit all the valid information. | Election will be deleted to the selected category. | Successful |
| Admin has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.9: Candidate Login Form

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Login Form** |
| Description | **This scenario covers the functionality of the login form for registered candidate.** |
| Module Name | **Candidate Login** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **21:10 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Candidate\_Login | Candidate Login | Candidate must have input all the valid information. | Successful login. | Successful |
| Candidate has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

Table 6.10: Voter Login Form

|  |  |
| --- | --- |
| Test Case | |
| Test scenario name | **Login Form** |
| Description | **This scenario covers the functionality of the login form for registered Voter.** |
| Module Name | **Voter Login** |
| Status | **Created** |
| Tester Information | |
| Name of Tester | **Aditi Mantri, Anshul Chhabra, Ashrami Shukla** |
| Date of Test | **6th April 2016** |
| Time of Test | **18:21 Hours** |
| Operating System | **Windows 7** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Case** | **User Input** | **Expected Result** | **Test Result** |
| Voter\_Login | Voter Login | Voter must have input all the valid information. | Successful login. | Successful |
| Voter has input some invalid information. | The error messages will be displayed to the user on those check fields. | Successful |

**CHAPTER 7**

**FUNCTIONAL & NON FUNCTIONAL REQUIREMENTS**

* 1. **PERFORMANCE REQUIREMENT**

There might be many users accessing to the web server simultaneously. As an online e-voting tool performance shouldn’t be affected much and response time for submitted page should be less than a minute.

* 1. **RELIABLITY**

The system should be reliable. Security is a major concern for an e-voting system. Process used in this system should be secure enough to be able to meet the requirements mentioned for e-voting. It requires database connections and network connections. Changes can be done in the databases to store the votes. All changes needs to be confirmed and if the transfer is complete the confirmation should be displayed. The changes should be monitored.

**7.3 RESPONSE TIME**

Response time is the total amount of time it takes to respond to a request for service. That service can be anything from a memory fetch, to a disk IO, to a complex database query, or loading a full web page. Ignoring transmission time for a moment, the response time is the sum of the service time and wait time.

**7.4 ROBUSTNESS**

Robustness is the ability of a computer system to cope with errors during execution. Robustness can also be defined as the ability of an algorithm to continue operating despite abnormalities in input, calculations, etc. Robustness can encompass many areas of computer science, such as robust programming, robust machine learning, and Robust Security Network form, the more robust the software. Formal techniques, such as fuzz testing, are essential to showing robustness since this type of testing involves invalid or unexpected inputs. Alternatively, fault injection can be used to test robustness

* 1. **SCALABLITY**

There is always a scope for improvement in eLearning. Much advancement can be made. An

option for video tutorials can be given, option for showing the latest trending posts and

technologies can be given, and an option for group chat and like button on eBooks’ and

tutorials can also be given. Every project has some future scope, the changes that can be done to make it better. Thus, this project is scalable i.e. is has some scope for future enhancements.

* 1. **SECURITY**

The voting process is very important and secret process as whole democratic administration is centralized on voting so the genuineness of votes and voters should be maintained at the highest priority to make the voting process fair and effective. Authentication, scalability, speed and accuracy these are few other important criteria’s to be satisfied by a typical voting system .Authentication can be considered as the most critical issue among all the above mentioned criteria. As the role of online voting system is crucial, it is difficult to come up with a system which is highly secure & accurate in all senses. If these factors of security are satisfied, then electronic voting could be a great improvement over paper systems.

**7.7 STABILITY**

Stability of service provider will depend upon number of user and server. If server will able to handle more users then stability will increase. Indirectly we can say that stability is proportional to the server capacity. Other than this if more features will require in future then we can easily add in this. It will not be able to hack so it will be more stable.

* 1. **SUPPORTABILITY**

It will be supportable for all type of user and every user have different priority and on the basis of that priority they have different features. It will supportable to every device. This web application (Nukem Global Technologies) will be supportable by every type of hardware and software but that should be supportable.

**7.8 TESTABILITY**

Software testability is the degree to which a software artefact (i.e. a software system,

software module, requirements- or design document) supports testing in a given test context.

If the testability of the software artefact is high, then finding faults in the system by means of

testing is easier.

Testability is not an intrinsic property of a software artefactand cannot be measured directly (such as software size). Instead testability is an extrinsic property which results from interdependency of the software to be tested and the test goals, test methods used, and test resources (i.e., the test context).

A lower degree of testability results in increased test effort. In extreme cases a lack of testability may hinder testing parts of the software or software requirements at all. In order to link the testability with the difficulty to find potential faults in a system (if they exist) by testing it, a relevant measure to assess the testability is how many test cases are needed in each case to form a complete test suite (i.e. a test suite such that, after applying all test cases to the system, collected outputs will let us unambiguously determine whether the system is correct or not according to some specification). If this size is small, then the testability is high. Based on this measure, a testability hierarchy has been proposed.

**7.10 FAULT TOLERANCE**

Fault tolerance is the property that enables a system to continue operating properly in the event of the failure of (or one or more faults within) some of its component. So this web application will be fault tolerance because it has different-different modules so if one module gets fail then it cannot be affect another module.

**CHAPTER 8**

**CONCLUSION**

**8.1. CONCLUSION**

Our proposal enables a patient to request his/her demand through internet without using any traditional way thus this system is easy for staff member also because the requested service will not go through to many stages , it will be directly received by the department head and he/she is responsible for allotting the staff. After completion of staff patient can provide proper feedback .

Hospital staff has the facility to study the services used by the patient graphically ,so that they can easily improve their management when their will be rush.

Patient’s family member don’t have to go to pharmacy they can order that online and bill will be generated according to that in this case pharmacist will receive a file according to the patient id and in that file he has the list of medicines ordered by the patient .

**8.2. FUTURE ENHANCEMENT**

We have a future perception of making our web application multilingual. Increasing its security so that this could be used for reports also .

* Making this web drive application multilingual.
* Security of the database will be improvised
* Reports will be provided on email.
* Patient id will not be removed.
* Making real time server.

**APPENDICES**

**APPENDIX A:**

**USER MANUAL**

1. The receptionist will have to register for patient
2. Patient have to login.
3. Patient can order medicine , order meal or request for particular service.
4. Then that request will be directly received by the department head.
5. Department head will allot staff for requested service after login with department id.
6. After that patient will click on complete request and that staff member will be free for further request.
7. Patient can also provide feedback for service.
8. Bill will be generated by the website automatically by the services used by the patient.
9. Admin can view the requested service graphically which will help them to manage staff member i.e how many member they want in particular department and at what time.

**GLOSSARY**

Admin : The administration of hospital.

Authenticate : Something which is true, genuine, or valid

Patient : A person who is admit in hospital.

Department head : A person who is the head for particular department in the hospital.

Staff member : A person who works in the hospital for particular duty.

Receptionist : A person who register for patient and is always available.

**APPENDIX B:**

**REFERENCES**

1. Available at http://www.simplyvoting.com/
2. Available at https://www.polleverywhere.com/
3. Available at https://en.wikipedia.org/wiki/Electronic\_voting
4. Available at <https://www.electionsonline.com/>
5. Available at <http://www.smartmatic.com/voting/electronic-voting/>
6. Available <http://www.ijarcsse.com/docs/papers/Volume_3/1_January2013/V3I1-0221.pdf>
7. Available at <http://lorrie.cranor.org/voting/sensus/ssp/node11.html>
8. Available at <https://en.wikipedia.org/wiki/Two-round_system>
9. Available athttps://en.wikipedia.org/wiki/Ranked\_voting\_system

**ABBREVIATIONS**