**CHAPTER 1**

**INTRODUCTION**

**1.1 Introduction**

There are numerous websites available that allow online learning over the Internet. The purpose of this project is to implement a website that will allow users with an internet connection to engage themselves in online learning. The development of this project is centered on the development of a website that would allow proper log in / sign up of users, search material, download file, chatting platform between teachers and students.

**1.2 Problem Introduction**

This project is to create a website with a server and clients to enable the clients to study online. This project is to simulate the interface between a teacher and a student who are enrolled or signed up on this website. Here, not only a student can prepare for his / her academics, placements or GATE but can also clear doubts by interacting with the teachers available online i.e. a chatting platform created here. Also, the student is given a platform to download resource / study material and also to comment for the doubts.

**1.3 Innovative Idea of Project**

* **Digital Library** that provides all the learning materials for the students. It should include notes and questions for each topic.
* **Video Lectures** of the subjects are available as well as can be downloaded by students. This can be viewed by all the registered students only.
* **Forum** is a platform where students can discuss their doubts with teachers.

**1.4 Project Objective**

The project is aimed at providing a platform to all Engineering aspirants to study at single place without searching for different platforms for learning academics, competitive and placement materials. This website “ WALK2CAREER “ is one such platform which will provide its users the single gateway to start preparing for multiple exams at a single platform, with a promise of being guided by reputed faculties from reputed colleges.

**1.5 Scope of Project**

The proposed E-Lerning system is simple to use for both the Teachers and students. The proposed system includes the following:

**Web – based access**

All users who have Internet facility can access the system wherever and whenever they wish.

**Document management**

System allows downloading of documents (study material & video lectures).

**Chatting platform**

Here the students can discuss their problems, doubts or can ask for any career related guidance from teachers available on the website.

**1.6 Related Previous Work**

The World Wide Web (www or web) E-Learning provides a simple ‘point and click’ means of exploring the immense volume of pages of information residing on the Internet. Information on Web is presented on Web pages, which appear as a collection of text, graphics, pictures. In addition, a Web page can contain hyperlinks to other Web pages. Much of the web’s success is due to the simplicity with which it allows users to provide, use and refer to the information distributed geographically around the world. This inturn results in time consuming system where users get confused which link to follow and which link not to follow. So, to overcome these issues, we have come up with a single and precise platform to help users find their answers.

**1.7 Project Perspective**

The system to be developed here is an E-Learning website named “ WALK2CAREER “. It is a centralized system. It is a website with centralized database server. All the users get access to the website by logging in to the website.

There is a two way communication between the registered teachers and students. This website can be used for studying, discussion and downloading study material for three main purposes that are Engineering academics, GATE preparation and Placement related data. It allows users to find other logged in users. This website can be used for surfing along with one to one conversation. User download the files for getting material to study. This website can be used for video lectures download too. The user comprises of the teachers and students where they can interact with each other by a chatting platform. This chat application can be used for message sharing. Also logged in user gets a interface where his profile is created where he can scroll and see activities going on website.

**CHAPTER 2**

**SOFTWARE DEVELOPMENT MODEL**

SDLC, Software Development Life Cycle is a process used by software industry to design, develop and test high quality software. The SDLC aims to produce high quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.



Figure 2.1 Software Development Life Cycle Stages

A typical Software Development life cycle consists of the following stages:

**Stage 1: Planning and Requirement Analysis**

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry.

This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational, and technical areas. Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

**Stage 2: Defining Requirements**

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through „SRS‟ Software Requirement Specification document which consists of all the product requirements to be designed and developed during the project life cycle.

**Stage 3: Designing the Product Architecture**

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS Design Document Specification. This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints , the best design approach is selected for the product. A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minute of the details in DDS.

**Stage 4: Building or Developing the Product**

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle. Developers have to follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers etc are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java, and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

**Stage 5: Testing the Product**

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However this stage refers to the testing only stage of the product where products defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

**Stage 6: Deployment in the Market and Maintenance**

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometime product deployment happens in stages as per the organizations‟ business strategy. The product may first be released in a limited segment and tested in the real business environment (UAT User acceptance testing). Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

**CHAPTER 3**

**PROJECT PLANNING**

Planning is a critically important and required step in any successful project for system and software development. The objective of the project planning process is the development of a Baseline Project Plan (BPP) and the Statement of Work (SOW). The BPP becomes the foundation for the remainder of the development system. The SOW produced by the team clearly outlines the objectives and constraints of the project. During the planning of the software products, the scope of the software is described, various risk factors, cost and schedule are analyses and resources required are established.

**3.1 Project Planning**

Once a project is found to be feasible, software project managers undertake project planning. Project planning consists of the following activities:

1. Project scope

2. Project schedule

3. Project team organization

4. Technical description of proposed system

5. Project standards, procedures and proposed techniques tools

6. Quality assurance plan

7. Configuration management plan

8. Documentation plan

9. Data management plan

10. Resource management plan

11. Test plan

12. Training plan

13. Security plan

14. Risk management plan

15. Maintenance plan

**3.1.1 Project Scope**

It involves estimating some basic attributes of the project like:

**Cost:** How much will it cost to develop the project?

Cost of the Accessible & Server project is nil.

**Duration:** How long will it take to complete the development?

Project took 4 months of active development.

**Effort:** How much effort would be required?

The project required extra ordinary efforts for both the designing and the coding.

**3.1.2 Project Schedule**

Scheduling the project task is an important project planning activity. It involves deciding which task would be taken up when. In order to schedule the project activities a project manager needs to do the following:

1. Identify all the tasks needed to complete the project.

2. Break down large tasks into small activities.

3. Determine the dependency among different activities.

4. Establish the most likely estimates for the time durations necessary to complete the activities.

5. Allocate resources to activities.

6. Plan the starting and ending dates for various activities.

7. Determine the critical path. A critical path is the chain of activities that determines the duration of the project.

**3.1.3 Project Team Organization**

It includes staff organization and staffing plans that means that each person is given his role in project development according to his skills.

The project team members along with their roles are:

1. Abhay Singh (1313310007)-Coding and Testing

2. Gopal (1313310057) - Coding and Testing

3. Ritika Singh (1313310122) - Designing and Testing

4. Shallu Dwivedi (1313310137) - Designing and Documentation

5. Sonal Gupta (1313310166) – Designing and Documentation

**3.1.4 Risk Management Plan**

A risk is something that may happen and if it does, will have an adverse impact on the project. The project did not require any risk management as no real risk was involved.

**CHAPTER 4**

**REQUIREMENT ANALYSIS**

The software requirement analysis activity is the next step of the software project planning in the software development. This phase is followed by software design. So, basically this phase is the intermediate task to be performed in between the computer system engineering and software design .The key and primary objectives of the requirements in terms of relationships, provide a basis for the design and provide a basis for validation for the software after it is built.

**4.1 Software Requirement Specification (SRS)**

**4.1.1 Introduction**

1. Background: The project is inspired with a view to understand the working & design of a website, chatting interface and file downloading over internet.

2. Purpose: To share study material in the form of files and video lectures along with chatting facility between the teachers and students.

3. Scope: The scope of the project covers all the science behind creating a programming language. The proposed WALK2CAREER system is simple to use for both the teachers and the students. The proposed system includes the following:

Web – based access: All users who have Internet facility can access the system wherever and whenever they want.

Document management: System allows downloading of documents (notes and video lectures ).

Chatting platform: It provides a chatting interface between the users i.e. the teachers and students.

4. Intended audience: Users having an account on WALK2CAREER website.

**4.1.2 Overall Description**

1. Product perspective: Independent programming language and its compiler software are developed in the project.

2. User classes & characteristics:

The users of the system include students, teachers and administrator.

Students:

* Students can easily access learning materials from anywhere at any time.
* Notes and video lectures are also available.
* Students can also discuss with the Teachers to clear their doubts.

Teachers / Mentors:

* Discussion forum is also provided for the users for expressing their views.
* Teachers can also download text and lectures.

Administrator:

* Administrator can upload some material to viewer.
* Administrator can also provide the uploaded material for downloading.

3. Operating Environment: Microsoft Window

**4.2 Functional Requirements**

1. **Registration:**

* Only registered users should be able to access the resources in the system.
* Administrator must be able to add or delete new topics to the system.
* Administrator must be able to add or delete users to the system.

1. **Resources:**

* Teachers must be able to download documents, pdfs, and video lectures from the system.
* Students must be able to view and download the resources.
* It should alow students to view documents, pdfs, video lectures.

1. **Communication**

* Students must be able to communicate with Teachers to clear doubts.

**4.3 Non - Functional Requirements**

1. User Interface:

a) The system shall maintain an easy to use interface across all functionality and for all users.

b) The clients user interface should be compatible with all commonly used browsers, such as Internet explorer, Firefox, Google chrome and Safari.

2. Scalability:

The system shall be able to scale based on the number of users using the system.

3. Security:

a) The administrative system should be protected from unauthorized access.

b) The data storage should protected from attacks and unauthorized access.

c) The interface should be protected from attacks.

4. Portability:

a) The system should run on a variety of operating systems that support the Java language. b) The system should run on a variety of hardware.

5. Maintainability:

a) The system should be easy to maintain.

b) There should be a clear separation of HTML and Java interface code.

c) There should be a clear separation between the interface and the business logic code.

6. Exception handling:

a) Exceptions should be reported effectively to the user if they occur.

**4.4 Technology specification**

**Tools Used:**

Application architecture – JAVA, J2EE

**4.4.1 JAVA**

Java is an object oriented programming language developed by Sun Microsystems a company best known for its high end UNIX workstations. Java language was designed to be simple, small and portable across platforms, operating systems, both at the source and at the binary level, which means that Java programs ( applet and application ) can run on any machine that has the Java virtual machine ( JVM ) installed.

**4.4.2 J2EE**

Java Platform, Enterprise Edition or Java EE is a widely used platform for server programming in the Java programming language. The Java platform ( Enterprise Edition ) differs from the Java Standard Edition Platform ( Java SE ) in that it adds libraries which provide functionality to deploy fault tolerant, distributed, multi-tier Java software, based largely on modular components running on an application server.

**4.4.3 Glash Fish**

Glash Fish is an open source software implementation of the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed under the Java Community Process.

Glash Fish is developed in an open and participatory environment and released under the Glash Fish License version .glash fish is intended to be a collaboration of the best of breed developers from around the world.

**4.4.4 Development Tool- Net Beans**

Net Beans is a multi-language software development environment comprising a base workspace and an extensible plug in system for customizing the environment. It is written mostly in Java. It can be used to develop applications in Java and, by means of various plug-ins, other programming languages including Ada, C, C++, COBOL, Fortran, Haskell, Perl, PHP, Python, R, Ruby ( Ruby on Rails Framework ), Scala, Clojure, Groovy, Scheme, and Erlang. It can also be used to develop packages for the software Mathematic. Development environments include the NetBeans Java development tools ( JDT ) for Java and Scala, NetBeans CDT for C / C++ and NetBeans PDT for PHP, among others.

**4.4.5 Database Platform – DB2**

DB2 Database is the database management system that delivers a flexible and cost effective database platform to build robust on demand business applications and supports for J2EE and web service standards.

**4.5 Software Interface**

* Client On Intranet

Web Browser, Operating System (any)

* Web Server

Apache Tomcat, Operating System ( any )

* Database Server

DB2, Operating System ( any )

* Development End

NetBeans ( J2EE, Java, Java Bean, Servlets, HTML, XML, AJAX ), DB2, OS ( Windows ), Glash Fish ( Web Server )

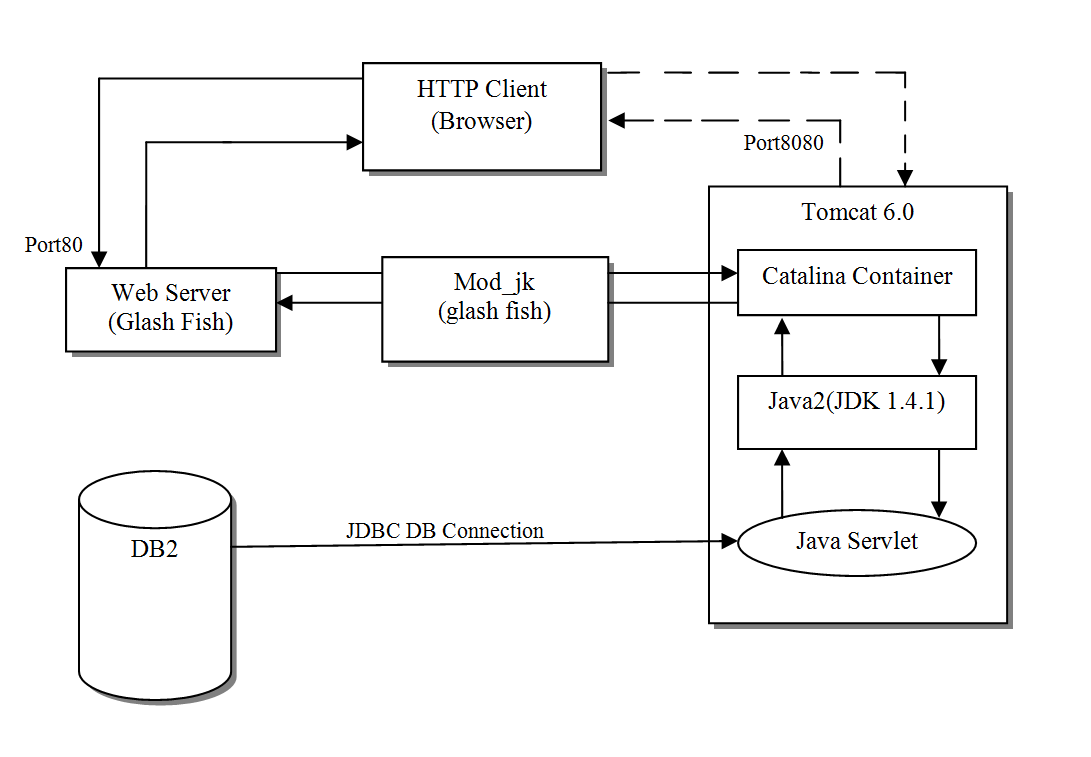


Figure 4.1 Pictorial Representation of Web Server

**CHAPTER 5**

**DESIGNING & ANALYSIS**

**5.1 System Overview**

The complete overview over the system can be seen as the clients can connect to each other, as well as connect to the server and communicate through this. The server runs as a standalone applications but can be controlled by the administrator. User information and information about banned IP addresses is stored in the database.

**5.2 System Design**

**5.2.1 UML Diagrams**

Activity Diagrams are a loosely defined diagram technique for showing workflows of stepwise activities and actions, with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

In the Unified Modelling Language (UML), a class diagram is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attricutes and the relationships between the classes.

**5.2.2 Use Case Diagram**

The primary actors of the system are teachers, students and administrator.

Diagram

**Documentation of Use Case Diagram:**

**i. Teachers / Mentors**

User role: Tutor

User functions:

a. content creation,

b. view learner’s progress,

c. answering students questions.

**ii. Students**

User role: Learner

User functions:

a. view learning materials,

b. view personal progress,

c. interactive discussions.

**iii. Administrator**

User role: Publisher

User functions:

a. upload some information

b. provide some materials

**5.2.3 Class Diagram**

**Documentation of Class Diagram:**

**Admin class:**

Main class in an e-learning system that is, WALK2CAREER is admin. Username, password / id and designation or email are the main attributes of this class. admin\_user() is the method for user login. registration\_user() is used for checking whether the username and password are valid. loging\_user() is for logging in and valid\_logging\_out() is for logging out from the system.

upload\_file() is the method for uploading files.

**Student Class:**

It has many attributes including username, password, usn no, email id, mobile no. Download() method is used to view or download the files or materials from the database. Search() method is used to searchthe particular information by selecting some topic / subject.

**Teacher Class:**

It has many attributes including username, password, college id. Download() method is used to view ordownload the files or material from the database. Search() method is used to search a particular information by selecting some topic.

**5.2.4 Sequence Diagram**

**a) Sudent**

**b) Professor**

**5.3 ER Diagram**

**E R Diagram of WALK2CAREEr learning system**

**5.4 Activity Diagram**

System Activity Diagram: User Registration Diagram

Activity Diagram: User Login

**Documentation of Activity Diagram**

The user logs in to the system using their username and password. Then they select a topic and views the learning materials. During learning, one can use the chat facility to clear doubts. Then they can take test by selecting the questionaire section.

**5.5 Data Flow Diagram**

**Introduction**

**DFD** is very useful in understanding a system and can be effectively used during analysis. A DFD shows the flow of data through a system. It views a system as a function that transforms the inputs into desired outputs. Any complex systems will not perform this transformation in a single step, and a data will typically undergo a series of transformations before it becomes output. The DFD aims to capture the transformations that take place within a system to the input data so that eventually the output data is produced. The agent that performs the transformation of data from one state to another is called a process (or a bubble). So a DFD shows the movement of data through the different transformations or the processes in the system. Named circles show the process and named arrows entering or leaving the bubbles represent data flows. A rectangle represents a source or sink and is a net originator or consumer of data. A source or sink is typically outside the main system of study.

DFD WITH DETAILED EXPLANATION

Context Level DFD

Level 0 DFD

Student

Level 1 DFD

Process 1(upload files)

Process2(view files)

Level 2 DFD

Process 3 (Select topic)

Process 4 (uploading to server)

**CHAPTER 6**

**CODING**

Coding is the most logical and dynamic part of the software development process. This phase of the software life-cycle is concerned with the development of code that will implement the design. This code is written in a formal language called a programming language. Programming languages have evolved over time from sequences of ones and zeros directly interpretable by a computer, through symbolic machine code, assembly languages, and finally to higher-level languages that are more understandable to humans. The design is translated into the code using different languages available for this purpose. The coding phase is concerned with:

**6.1 Front-end Coding**

Front-end development is concerned with the development of the user-interface. The user interface is referred to the view that exposes the software to the user and the user interacts with the software using the same interface. The web pages are created using MySQL and further customized using other designing tools such as eclipse to give an appeal to the website. The simple pages having text, forms for providing information, media content and many other entities are provided through the web pages. This part of coding affects a normal user the most as the visuals are most visible to the users. In all, a front-end developer is responsible for the interior design of a house that’s been built by a back-end. The frontend of an application is distinctly human. It’s what the user sees, touches and experiences. In this respect, empathy is a required characteristic of a good frontend developer. The frontend of an application is less about code and more about how a user will interpret the interface into an experience.

**6.2 Programming Language for Development**

Given the nature of the project and the complexities involved in it, ANSI C and the assembly language were chosen for its development. The main focus during the development has been:

1. Efficiency

2. Low Space Complexity

3. Low Time Complexity

4. Code Portability

The whole project has been developed in Java Code and Eclipse.

**6.3 Coding Standards**

The following coding standards have been maintained:

1. No use of clever or tricky programming.

2. No use of identifier for multiple purposes.

3. Variable must be given a meaningful name.

4. The code should be well documented.

**6.3.1 Coding Convention**

The following coding convention has been used throughout the project:

1. Functions – first letter is small

2. Macros - all letters are capital

**6.4 List of Source Code Files**

The table 6.1 shows the modules along with the source file names which they have been developed in.

Table 6.1 – List of Source Code File

|  |  |
| --- | --- |
| **File Name(s)** | **Implementation Module** |
| MainActivity.java | Main module |
| ChatCenter.java | Chat module |
| LoginActivity.java | Login module |
| MyChatService.java | Text module |

**6.5 Code Review**

Code review for the module is carried out after the module is successfully compiled and the syntax errors are eliminated. Normally two types of reviews are carried out:

1. Code Walkthrough

2. Code Inspection

**6.5.1 Code Walkthrough**

It is an in formal code analysis technique. In this technique, after the module has been coded, it is successfully compiled and all the syntax errors are eliminated. The main objectives of code walkthrough are to discover the algorithmic and logical errors.

**6.5.2 Code Inspection**

In contrast of code walkthrough, the aim of code inspection is to discover some common types of errors caused due to oversight and improper programming. Following is the list of classical programming errors checked during code inspection:

1. Use of uninitialized variable.

2. Jumps into loops.

3. Non-terminating loops.

4. Improper storage allocation and reallocation.

6. Mismatches between actual and formal parameter in procedure calls.

7. Use of incorrect logical operators or incorrect precedence among operators.

**CHAPTER 7**

**TESTING**

**7.1 Life Cycle Models**

The stage of planning the development process involves defining, developing, testing, deliver, operate, and maintain a software product. Different lifecycle models emphasize different aspects of the lifecycle and no single lifecycle model is suitable for all software products. A lifecycle model that is understood and accepted by all concerned parties improves project communication and enhances project manageability, resource allocation, cost control, and product quality.

The Phased Life Cycle Model -The phased lifecycle model represents software life cycle as a series of successive activities. Each phase requires well-defined input information, utilizes well-defined processes and results in well-defined products. The phased model consists of following phases.

* Analysis
* Design
* Implementation
* System Testing and Maintenance

This model is sometimes called the Waterfall Chart, the implication being that the products cascade from one level to another in smooth progression.

Table 7.1 Life Cycle Model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Analysis | Design | Implementation | System Testing | Maintenance |
| Planning,  user needs  definition | Design  details | Code, debug  & test | Integration  And acceptance | Enhance, Fix  & adapt |

The Analysis Stage consists of Planning and Requirements definition Major include understanding the customer’s problem, performing a feasibility study, developing a recommended solution strategy, determining the acceptance criteria and planning development process. The products of planning are a System definition and a project plan. The Software Design follows analysis. Design is concerned with its software components, specifying relationships among components specifying some structure, maintaining a record of design decisions and providing blueprint implementation phase. Design consists of detailed design and Architectural design.

The implementation phase of software development involves translation design specification into source code, and debugging, documentation and unit testing the source code. To enhance the quality of the software the methods are structured control constructs, built in and user defined data types, secure type checking, flexible scope rules exception handling mechanism, concurrency constructs and separates compilation modules.

System Testing involves two kinds of testing integration testing and acceptance testing. Developing a strategy for integrating the components of a software system into a functioning whole requires careful planning so that modules are available for integration when needed. Acceptance testing involves planning and execution of various tests in order to demonstrate that the implemented system satisfies the requirements document.

The Maintenance phase comes after the acceptance by the customer and release of the system for production work. Maintenance activities include enhancements of capabilities, adaptation of software to new processing environments, and correction of software bugs.

This project follows the Phased Life Cycle Model or the Water Fall model to a large extent.

The analysis stage consisted of listening to the needs and requirements of all departments obtaining the required format of the system as desired by them, taking the required data to be stored for future use etc. In the design stage the structure of the system was designed and all the required screens were formatted. This was then shown to the medical officer’s approval and the system was built. Implementation phase was also done at as they provided a computer with all the required software and with required configuration. The coding and debugging was done even after this stage certain changes were made as made as requested by the guide. The testing was done to check for any errors or bugs or unwanted behaviour in the system. Individual modules as well as the whole system were tested separately.

**7.2 Software Testing**

Software Testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding, Testing presents an interesting anomaly for the software engineer. Testing Objectives include:

Testing is a process of executing a program with the intent of finding an error.

A good test case is one that has a probability of finding an as yet undiscovered error. A successful test is one that uncovers an undiscovered error .

* Testing Principles:

All tests should be traceable to end user requirements. Tests should be planned long before testing begins. Testing should begin on a small scale and progress towards testing in large. Exhaustive testing is not possible. To be most effective, testing should be conducted by an independent third party.

* Testing Strategies:

A Strategy for software testing integrates software test cases into a series of well-planned steps that result in the successful construction of software. Software testing is a broader topic for what is referred to as Verification and Validation. Verification refers to the set of activities that ensure that the software correctly implements a specific function. Validation refers he set of activities that ensure that the software that has been built is traceable to customer’s requirements.

* **Unit Testing:**

Unit testing focuses verification effort on the smallest unit of software design that is the module. Using procedural design description as a guide, important control paths are tested to uncover errors within the boundaries of the module. The unit test is normally white box testing oriented and the step can be conducted in parallel for multiple modules.

* **Integration Testing:**

Integration testing is a systematic technique for constructing the program structure while conducting test to uncover errors associated with the interfacing. The objective is to take unit tested methods and build a program structure that has been dictated by design.

* **Top-down Integration:**

Top down integrations an incremental approach to construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main control program. Modules subordinate to the main program are incorporated in the structure either in the breath-first or depth-first manner.

* **Bottom-up Integration:**

This method as the name suggests, begins construction and testing with atomic modules i.e., modules at the lowest levelling the program structure. Because the modules are integrated in the bottom up manner the processing required for the modules subordinate to a given level is always available and the need for stubs is eliminated.

* **Validation Testing:**

At the end of integration testing software ids completely assembled as a package. Validation testing is the next stage which can be defined as successful when the software functions in the manner reasonably expected by the customer. Reasonable expectations are those defined in the software requirements specifications. Information contained in those sections form a basis for validation testing approach.

* **System Testing:**

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated to perform allocated functions.

* **Recovery Testing:**

It is a system test that forces the system to fail in a variety of ways and verities that the recovery is properly performed.

* **Security Testing:**

Attempts to verify the protection mechanisms built into the system.

* **Performance Testing:**

This method is designed to test runtime performance of software within the context of an integrated system. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing is the exposure of the system to trial input to see whether it produces correct output.

* **Testing Phases:**

Software testing phases include the following:

Test activities are determined and test data selected. The test is conducted and test results are compared with the expected results.

There are various types of Testing:

* **Unit Testing:**

Unit testing is essentially for the verification of the code produced during the coding phase and the goal is test the internal logic of the module/program.

This project is thoroughly tested by exposing it to the various test cases regarding correct event generation, as this project passed all the tests its quality is completely assured.

* **Integration Testing:**

All the tested modules are combined into sub systems, which are then tested. The goal is to see if the modules are properly integrated, and the emphasis being on the testing interfaces between the modules. On this project integration testing is done mainly while implementing menus in a sample application such as Browser for Mobiles.

* **System Testing:**

It is mainly used if the software meets its requirements. The reference document for this process is the requirement document.

* **Acceptance Testing:**

It is performed with realistic data of the client to demonstrate that the software is working satisfactorily.

**7.3 Testing Methods:**

Testing is a process of executing a program to find out errors. If testing is conducted successfully, it will uncover all the errors in the software. Any testing can be done basing on two ways:

* **White Box Testing:**

It is a test case design method that uses the control structures of the procedural design to derive test cases. Using this testing a software Engineer can derive the following test cases:

Exercise all the logical decisions on either true or false sides. Execute all loops at their boundaries and within their operational boundaries. Exercise the internal data structures to assure their validity.

* **Black Box Testing:**

It is a test case design method used on the functional requirements of the software. It will help a software engineer to derive sets of input conditions that will exercise all the functional requirements of the program. Black Box testing attempts to find errors in the following categories:

-Incorrect or missing function

-Interface errors

-Errors in data structures

-Performance errors

-Initialization and termination errors

**7.4 Test Cases:**

Table 7.2 – Test Cases

**CHAPTER 8**

**IMPLEMENTATION**

**8.1 System implementation**

Implementation is the stage of the project where the theoretical design is turned in to a working system. The implementation state in a system project is its own right. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, training of staff in the changeover procedure and evaluation of change over methods.

Once the planning has been completed, the major efforts are to ensure that the program in the system is working properly.

At the same time concentrate on training user staff. When the staff has been trained a full system can carry out.

The various activities involved while implementing a project:-

-End user education and training.

-Training on application software.

-System testing

-Parallel run and change over to new system

-Post implementation review

**8.2 Home Page**

**8.3 Sign Up Page**

**8.4 Login Page**

**8.5**

**CHAPTER 9**

**CONCLUSION & FUTURE ENHANCEMENTS**

**9.1 Conclusion**

The primary application of the website is to gather as much registered students as it can.

This website is useful in the following ways:

The user can find multiple use of this website as it is single platform that gives access to all Engineering graduates for academic, GATE & Placement preparations.

* The user can download study materials and video lectures.
* There is a communication platform between a student and teacher where their doubts can be discussed

Last but not the least, this website, “WALK2CAREER” is a mixture of Information & Communication Technology, used in providing learning experiences to the students on-line through the use of Internet services and web technology of computers on the same lines as witnessed by us in the form of e-mail, e-banking, e-commerce, etc in our day to day life.

**9.2 FUTURE ASPECTS**

Login session information:

We can include a feature in this website that will keep a record of the user’s login session. Profile Database:

The profile information of every user can be stored at the server.

Login Timeout:

This feature allows the user to be logged in only for a specific time. After this time span ends, the user is automatically logged out.

File transferring and sharing:

User can transfer one or more files to other users. A file can also be shared between two or more users.

**REFERENCES**

1. “Java Technology”, http://java.sun.com/

2. “MySQL. The world’s most popular open source database”, http://www.mysql.com/

3. “A visual guide to layout managers”, <http://java.sun.com/docs/books/tutorial/uiswing/layout/visual.html>

4. Eric Freeman, Elisabeth Robson, *Head First HTML5 Programming*, O’Reilly, 2013.

5. Kathy Sierra, Bert Bates, *Head First Java*, O’Reilly, 2011.

6. Sommerville, Ian, *Software Engineering*, Boston: Pearson.