**CHAPTER 1**

**INTRODUCTION**

Today, Online Examination System is considered a fast developing examination method because of its accuracy and speed. It is also needed less manpower to handle the examination. Almost all organizations today, are managing their exams by online examination system, since it reduces student's time in examinations. Organizations can also easily monitor the progress of the student that they give through an examination. As a result of this, the result is calculated in less time. It also helps diminishing the need for paper. Online examination project in Java is very useful to learn it, According to today’s requirement Online examination system is significantly important to the educational institution to prepare the exams, saving the time and effort that is required to check the exam papers and to prepare the results reports. Online examination system helps the educational institutions to monitor their students and keep eyes on their progress. The best use of this system in Scholastic Institute and training centres because it helps in managing the exams and get the results in easy and an efficient manner. Until today the preparing for exams and preparing the results was performed manually, this required more time to complete.

Many different researches have focused on the subject of an online examination system these work can be represented as following. EMS manages the examination and auto-grading for student’s exams and supports conducting exams, collects the answers, auto mark the submissions, and produce the reports for the test. EMS supports secure login, multi-instructor, and portability features. However, the other features: resumption capability, random question selection, random questions distribution, and random choices distribution are missing. The project evaluates the examiners by using the online examination system concept. The exams will be totally customizable. This system will check results automatically basing on students answers. Online Examination System is a web-based online examination system developed to address issues such as lack of timing flexibility for automation candidates log-off upon expiration of allowed time, result integrity, guaranty, stand-alone deployment, need for flexibility, robustness, designed to support the examination processes and overcome challenges framing the conduct of examination, auto- marking, auto- submission, and generation report of examination result.

**CHAPTER 2**

**SYSTEM ANALYSIS**

**2.1 Existing System:-**

Since the traditional have many drawbacks such as time consuming, Difficulty of analysing the test manually, More observers are required to take exam of many students, Results are not accurate since calculations is done manually, The chance of losing exam's result is higher in current systems, Checking of result is time consuming since it done manually, Limitation of no of student can give examination at a time. One more drawback in existing system is use of Maximum Paper work. As per new government rules , government is trying to decrease the paper work as many as possible considering environmental hazardless.

**2.2 Proposed System:-**

With the development of information technology and use it in an orderly and properly helps to overcome the existing error in the manual system. Online examination system saves the exams information in a database, and this make it an easier way to give exam teachers can add theirs exams rules, and student can give exam in a totally automated system. Online examination system save the exams information in a database , teachers can add/Edit/delete questions, set the correct answers, specify the exam period, register students ,delete students , show questions for students randomly, calculate and show the final results for students. The Online Exam System is beneficial for us in following sense:

* On-line Exam system is designed for Educational Institutes (like schools, universities, training centers).
* The system handles all the operations, and generates reports as soon as the test is finish, that includes name, mark, time spent to solve the exam.
* Allow students to see or display his answers after the exam is finish.
* The type of questions is only multiple choice or true and false.

**CHAPTER 3**

**SYSTEM CONFIGURATION**

**3.1 Hardware System Configuration**:-

* Processor - Pentium –III
* Speed - 1.1 Ghz
* RAM - 256 MB (min)
* Hard Disk - 20 GB
* Floppy Drive - 1.44 MB
* Key Board - Standard Windows Keyboard
* Mouse - Two or Three Button Mouse
* Monitor - SVGA

**3.2 Software System Configuration:-**

* Operating System : Windows95/98/2000/XP/7/8.1/10.1 etc.
* Front End : HTML, JSP, JavaScript.
* Database : MySQL Database.
* Database Connectivity : JDBC.

**CHAPTER 4**

**FEASIBILITY STUDY**

## Introduction:

A feasibility analysis involves a detailed assessment of the need, value and practicality of a p systems development. Feasibility analysis n forms the transparent decisions at crucial points during the developmental process as we determine whether it is operationally, economically and technically realistic to proceed with a particular course of action.

Feasibility analysis can be used in each of the steps to assess the financial, technical and operational capacity to proceed with particular activities.

### Types of feasibility:

A feasibility analysis usually involves a thorough assessment of the financial (value), technical (practicality), and operational (need) aspects of a proposal. In systems development projects, business managers are primarily responsible for assessing the operational feasibility of the system, and information technology (IT) analysts are responsible for assessing technical feasibility. Both then work together to prepare a cost–benefit analysis of the proposed system to determine its economic feasibility.

#### Operational feasibility:

A systems development project is likely to be operationally feasible if it meets the 'needs' and expectations of the organization. User acceptance is an important determinant of operational feasibility. It requires careful consideration of:

1. Corporate culture;
2. Staff resistance or receptivity to change;
3. Management support for the new system;
4. The nature and level of user involvement in the development and implementation of the system; direct and indirect impacts of the new system on work practices;
5. Anticipated performance and outcomes of the new system compared with the existing system;
6. Training requirements and other change management strategies; and
7. ‘Pay back’ periods (i.e. trade-off between long-term organisational benefits and short-term inefficiencies during system development and implementation).

#### General Feasibility:

A systems development project may be regarded as technically feasible or practical if the organization has the necessary expertise and infrastructure to develop, install, operate and maintain the proposed system. Organizations will need to make this assessment based on:

1. Knowledge of current and emerging technological solutions
2. Availability of technically qualified staff in-house for the duration of the project and subsequent maintenance phase;
3. Availability of infrastructure in-house to support the development and maintenance of the proposed system;
4. Where necessary, the financial and/or technical capacity to procure appropriate infrastructure and expertise from outside;
5. Capacity of the proposed system to accommodate increasing levels of use over the medium term;
6. The capacity of the proposed system to meet initial performance expectations and accommodate new functionality over the medium term.

**4.1 ECONOMICAL FEASIBILITY:**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### 4.2 TECHNICAL FEASIBILITY:

### 

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**4.3 SOCIAL FEASIBILITY:**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**Functional and Non-Functional Requirements:**

1. **Functoinal Requirements:**
2. **Inputs:**

* Administrator: Admin logins and handles Question Paper, Marking System, Student Approval.
* Student: Students performs various activities such as Registration, Approval, Appear for Examination, Check Result.

1. **Processing:**

All the information is stored in secure manner and is guarded.

**OUTPUT:**

Administrator stores the all the details such as Question Paper, Marking System, Examination, Student Result etc.. All registered Student can view the results information or details after login to the admin.

**Performance requirements**

Due to the high scope of the software, the performance requirements are high. The speed at which the software is required to operate is nominal. A processing rate of 5-10 seconds per query is acceptable.

1. **Error message design**

The design of error messages is an important part of the user interface design. As user is bound to commit some errors or other while designing a system the system should be designed to be helpful by providing the user with information regarding the error he/she has committed.

1. **Error detection:**

Even though every effort is make to avoid the occurrence of errors , still a small portion of errors are always likely to occur , these type of errors can be discovered by using validations to check input data.

The system is designed to be a user friendly one. In other words the system has been designed to communicate effectively with the user. The system has been designed with Button.

1. **Non Functional Requirements**

Performance is measured in terms of the output provided by the application.

Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely in the part of users of the existing system to give the requirement specifications because they are the people who finally use the system.

The requirement specification for any system can be broadly stated as given below:

* The system should be able to interface with the existing system.
* The system should be accurate.
* The system should be better than existing system.
* **Portability:** It should run on specified platforms successfully. To achieve this we should test the product on all platforms before launching the product. If our project runs successfully on different platforms then our system is portable in nature.
* **Reliability:** The system should perform its intended functions under specified conditions. If our system satisfies all the specified conditions then it is Reliable in nature.
* **Reusability:** The system should be extremely reusable as a whole or part. Make the system modularize and make sure that modules are loosely coupled. This project is having reusability nature because we can reuse whole or part of this project on other systems.
* **Robustness:** The system on the whole should be robust enough to perform well under different circumstances without any inconsistencies.
* **Testability:** The product of a given development phase should satisfy the conditions imposed at the start of that phase.
* **Usability:** It should be perfect and comfortable for users to work.
* **Security:** The system is completely based on the security. This system will provide security base on the password.

**CHAPTER 5**

**DESIGN ANALYSIS**

**5.1 UML Diagram:**

**5.1.1. UML Diagrams:**

UML is a method for describing the system architecture in detail using the blueprint.

UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. UML is a very important part of developing objects oriented software and the software development process. UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

**Definition:**

UML is a general-purpose visual modelling language that is used to specify, visualize, construct, and document the artefacts of the software system.

**UML is a language:**

It will provide vocabulary and rules for communications and function on conceptual and physical representation. So it is modelling language.

**UML Specifying:**

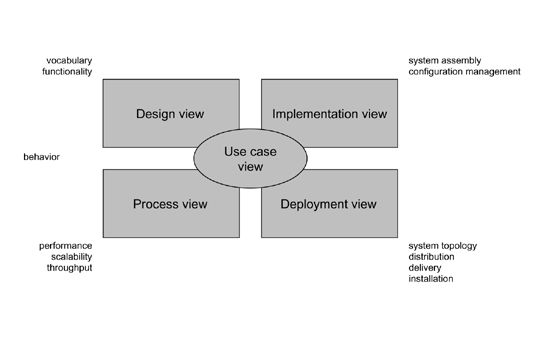
Specifying means building models that are precise, unambiguous and complete. In particular, the UML address the specification of all the important analysis, design and implementation decisions that must be made in developing and displaying a software intensive system.

**5.1.2 UML Visualization:**

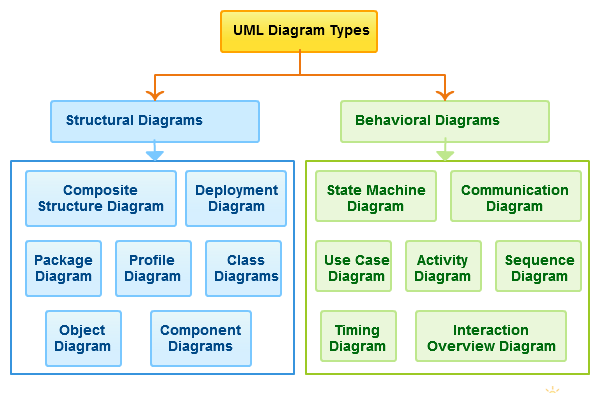
The UML includes both graphical and textual representation. It makes easy to visualize the system and for better understanding.

**5.1.3 UML Constructing:**

UML models can be directly connected to a variety of programming languages and it is sufficiently expressive and free from any ambiguity to permit the direct execution of models.



**Figure 5.1.1 Modeling a System Architecture using views of UML**

1. The use case view of a system encompasses the use cases that describe the behavior of the system as seen by its end users, analysts, and testers.
2. The *design view* of a system encompasses the classes, interfaces, and collaborations that form the vocabulary of the problem and its solution.
3. The *process view* of a system encompasses the threads and processes that form the system's concurrency and synchronization mechanisms.
4. The *implementation view* of a system encompasses the components and files that are used to assemble and release the physical system.The *deployment view* of a system encompasses the nodes that form the system's hardware topology on which the system executes. 

**Figure5.1.2 UML Diagram**

**5.1.4 Uses of UML :**

The UML is intended primarily for software intensive systems. It has been used effectively for such domain as

Enterprise Information System

1. Banking and Financial Services
2. Telecommunications
3. Transportation
4. Defense/Aerosp
5. Retails
6. Medical Electronics
7. Scientific Fields
8. Distributed Web

**Building blocks of UML:**

The vocabulary of the UML encompasses 3 kinds of building blocks

1. Things
2. Relationships
3. Diagrams
4. **Things:**

Things are the data abstractions that are first class citizens in a model. Things are of 4 types

Structural Things, Behavioral Things ,Grouping Things, An notational Things

1. **Relationships:**

Relationships tie the things together. Relationships in the UML are

Dependency, Association, Generalization, Specialization

1. **UML Diagrams:**

A diagram is the graphical presentation of a set of elements, most often rendered as a connected graph of vertices (things) and arcs (relationships).

There are two types of diagrams, they are:

Structural and Behavioral Diagrams

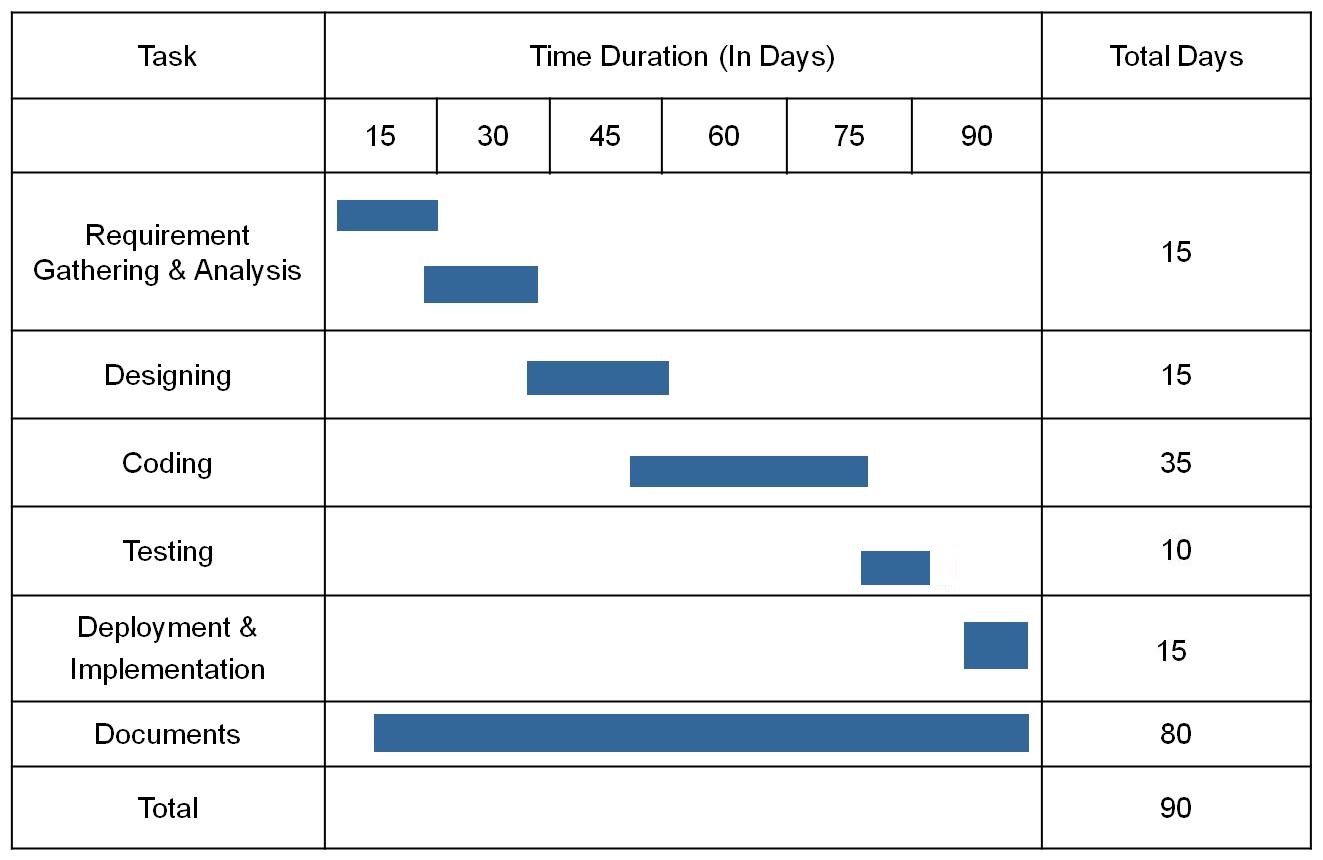
**5.2 Structural Diagrams:-**

The UML‘s four structural diagrams exist to visualize, specify, construct and document the static aspects of a system. ican View the static parts of a system using one of the following diagrams. Structural diagrams consists of Class Diagram, Object Diagram, Component Diagram, Deployment Diagram.

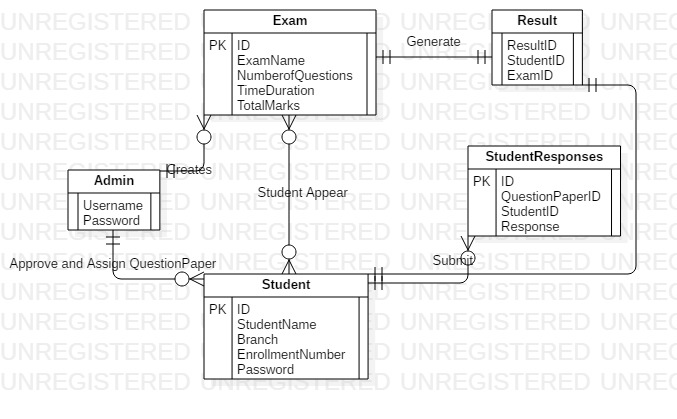
**5.3 Behavioural Diagrams :**

The UML’s five Behavioral Diagrams are used to visualize, specify, construct, and document the dynamic aspects of a system. The UML’s Behavioral Diagrams are roughly organized around the major ways which can model the dynamics of a system. Behavioral diagrams consists of Use case Diagram, Sequence Diagram, Collaboration Diagram, State chart Diagram, Activity Diagram.

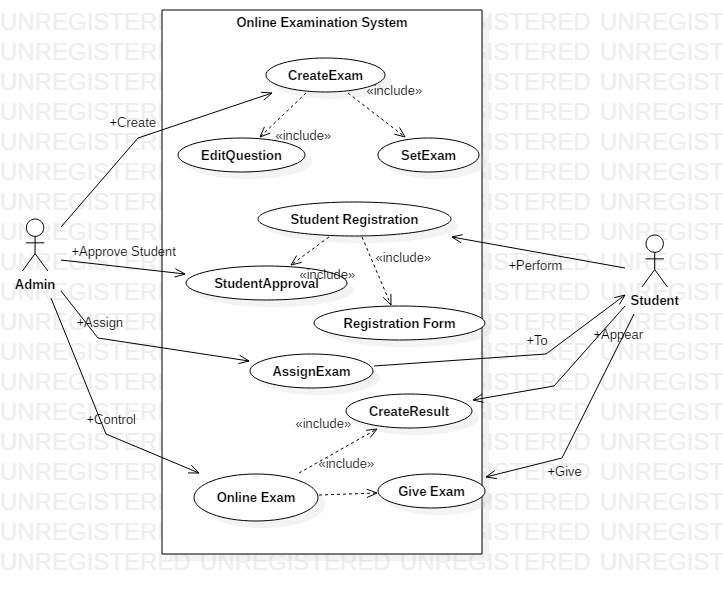
**5.4. Timeline chart**

****

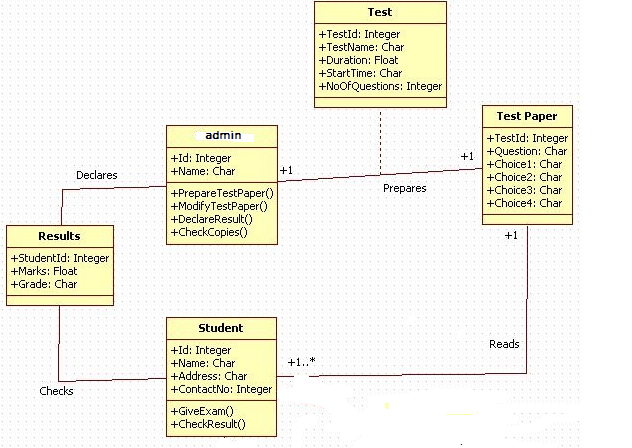
**5.5 E-R Diagram:-**



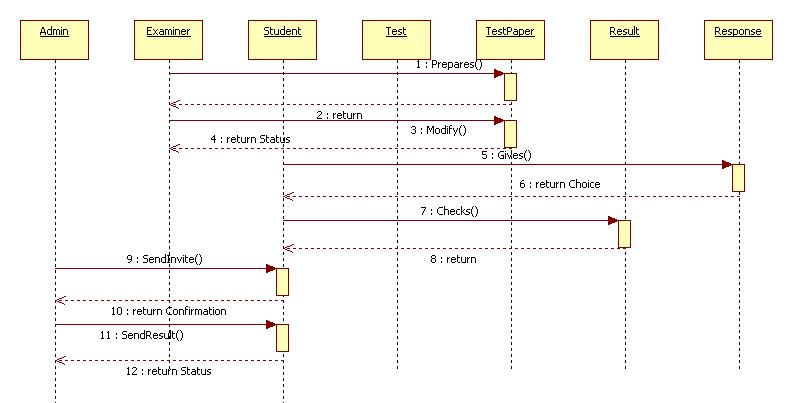
* 1. **Use case Diagram For Admin:-**



**5.7 Class Diagram:-**



**5.8 Sequence Diagram:**

****

**CHAPTER 6**

**TECHNOLOGIES USED**

**6.1 Introduction To Java:**

Java has been around since 1991, developed by a small team of Sun Microsystems developers in a project originally called the Green project. The intent of the project was to develop a platform-independent software technology that would be used in the consumer electronics industry. The language that the team created was originally called Oak.

The first implementation of Oak was in a PDA-type device called Star Seven (\*7) that consisted of the Oak language, an operating system called GreenOS, a user interface, and hardware. The name \*7 was derived from the telephone sequence that was used in the team's office and that was dialed in order to answer any ringing telephone from any other phone in the office.

Around the time the First Person project was floundering in consumer electronics, a new craze was gaining momentum in America; the craze was called "Web surfing." The World Wide Web, a name applied to the Internet's millions of linked HTML documents was suddenly becoming popular for use by the masses. The reason for this was the introduction of a graphical Web browser called Mosaic, developed by ncSA. The browser simplified Web browsing by combining text and graphics into a single interface to eliminate the need for users to learn many confusing UNIX and DOS commands. Navigating around the Web was much easier using Mosaic.

It has only been since 1994 that Oak technology has been applied to the Web. In 1994, two Sun developers created the first version of Hot Java, and then called Web Runner, which is a graphical browser for the Web that exists today. The browser was coded entirely in the Oak language, by this time called Java. Soon after, the Java compiler was rewritten in the Java language from its original C code, thus proving that Java could be used effectively as an application language. Sun introduced Java in May 1995 at the Sun World 95 convention.

Web surfing has become an enormously popular practice among millions of computer users. Until Java, however, the content of information on the Internet has been a bland series of HTML documents. Web users are hungry for applications that are interactive, that users can execute no matter what hardware or software platform they are using, and that travel across heterogeneous networks and do not spread viruses to their computers. Java can create such applications.

The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

* Simple
* Architecture neutral
* Object oriented
* Portable
* Distributed
* High performance
* Interpreted
* Multithreaded
* Robust
* Dynamic
* Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.



**Figure 6.1: Working Of Java**

You can think of Java bytecodes as the machine code instructions for the java virtual machine (Java VM). Every Java interpreter, whether it’s a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java bytecodes help make “write once, run anywhere” possible. You can compile your program into bytecodes on any platform that has a Java compiler. The bytecodes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.

**6.1.1 The Java Platform:**

A platform is the hardware or software environment in which a program runs. We’ve already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it’s a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

The java virtual mechine (Java VM)

The java application programming interface (Java API)

You’ve already been introduced to the Java VM. It’s the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as packages. The next section, What Can Java Technology Do?, highlights what functionality some of the packages in the Java API provide.

The following figure depicts a program that’s running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.



**Figure 6.1.1: The Java Platform**

Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time bytecode compilers can bring performance close to that of native code without threatening portability.

**6.1.2 Working Of Java:**

For those who are new to object-oriented programming, the concept of a class will be new to you. Simplistically, a class is the definition for a segment of code that can contain both data and functions.When the interpreter executes a class, it looks for a particular method by the name of **main,** which will sound familiar to C programmers. The main method is passed as a parameter an array of strings (similar to the argv[] of C), and is declared as a static method.

To output text from the program, iexecute the **println** method of **System. out,** which is java’s output stream. UNIX users will appreciate the theory behind such a stream, as it is actually standard output. For those who are instead used to the Wintel platform, it will write the string passed to it to the user’s program.

**6.2 Servlet:**

**Servlet** technology is used to create a web application (resides at server side and generates a dynamic web page).**Servlet** technology is robust and scalable because of java language. Before Servlet, CGI (Common Gateway Interface) scripting language was common as a server-side programming language. However, there were many disadvantages to this technology. We have discussed these disadvantages below. There are many interfaces and classes in the Servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse, etc.

## What is a Servlet?

Servlet can be described in many ways, depending on the context.

* Servlet is a technology which is used to create a web application.
* Servlet is an API that provides many interfaces and classes including documentation.
* Servlet is an interface that must be implemented for creating any Servlet.
* Servlet is a class that extends the capabilities of the servers and responds to the incoming requests. It can respond to any requests.
* Servlet is a web component that is deployed on the server to create a dynamic web page.

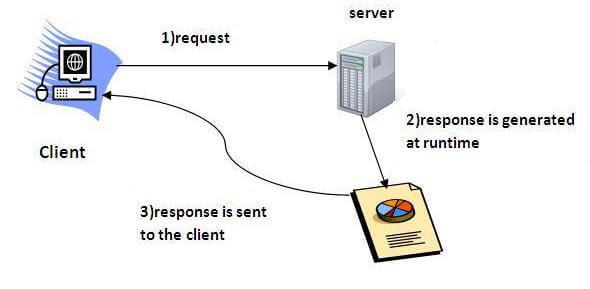


Fig6.2.1. Servlet Working

**6.3 Introduction To JSP:**

**JSP** technology is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc. A JSP page consists of HTML tags and JSP tags. The JSP pages are easier to maintain than Servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tags, etc.

### Advantages of JSP over Servlet

There are many advantages of JSP over the Servlet. They are as follows:

#### 1) Extension to Servlet

JSP technology is the extension to Servlet technology. We can use all the features of the Servlet in JSP. In addition to, we can use implicit objects, predefined tags, expression language and Custom tags in JSP, that makes JSP development easy.

#### 2) Easy to maintain

JSP can be easily managed because we can easily separate our business logic with presentation logic. In Servlet technology, we mix our business logic with the presentation logic.

#### 3) Fast Development: No need to recompile and redeploy

If JSP page is modified, we don't need to recompile and redeploy the project. The Servlet code needs to be updated and recompiled if we have to change the look and feel of the application.

#### 4) Less code than Servlet

In JSP, we can use many tags such as action tags, JSTL, custom tags, etc. that reduces the code. Moreover, we can use EL, implicit objects, etc.

### 6.2.2 The Lifecycle of a JSP Page

The JSP pages follow these phases:

* Translation of JSP Page
* Compilation of JSP Page
* Classloading (the classloader loads class file)
* Instantiation (Object of the Generated Servlet is created).
* Initialization ( the container invokes jspInit() method).
* Request processing ( the container invokes \_jspService() method).
* Destroy ( the container invokes jspDestroy() method).

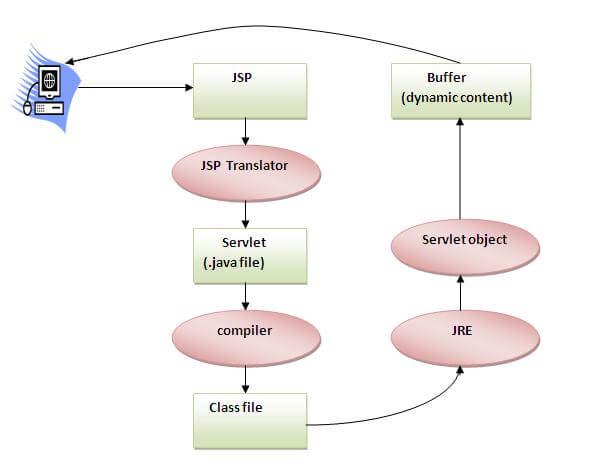


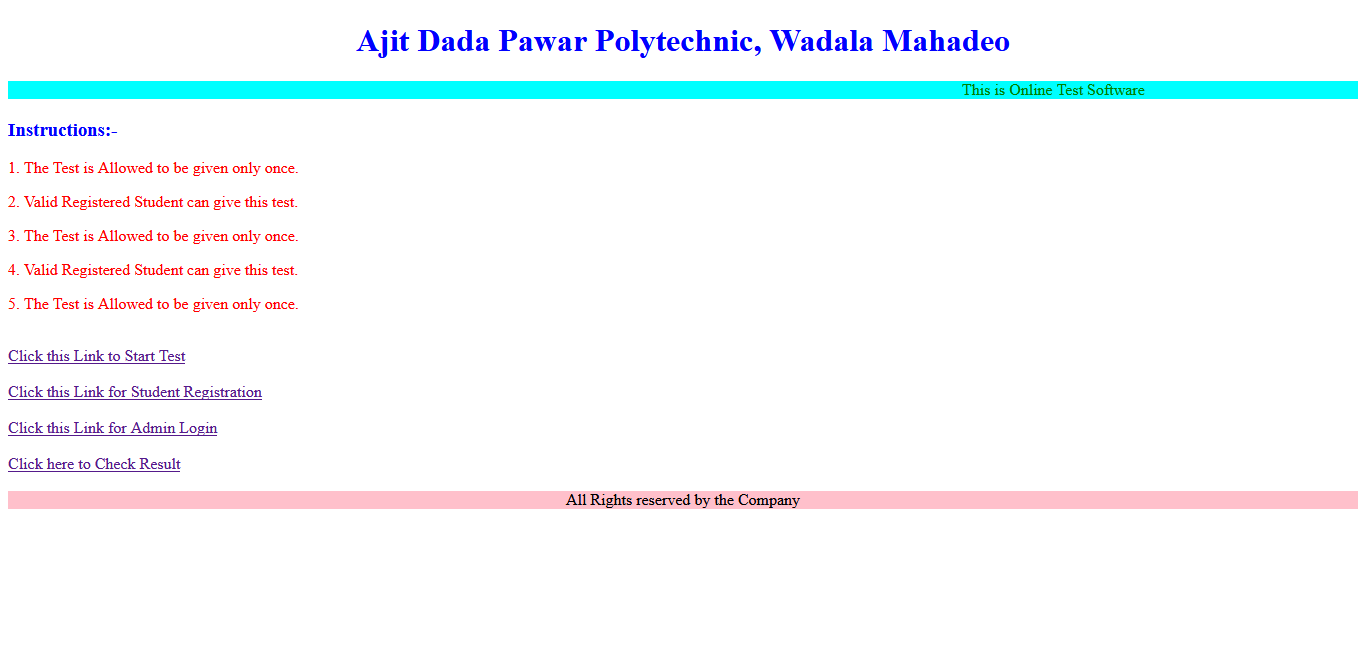
Fig.6.2.2JSP Life Cycle

As depicted in the above diagram, JSP page is translated into Servlet by the help of JSP translator. The JSP translator is a part of the web server which is responsible for translating the JSP page into Servlet. After that, Servlet page is compiled by the compiler and gets converted into the class file. Moreover, all the processes that happen in Servlet are performed on JSP later like initialization, committing response to the browser and destroy.

**CHAPTER 7**

**SCREENSHOTS**

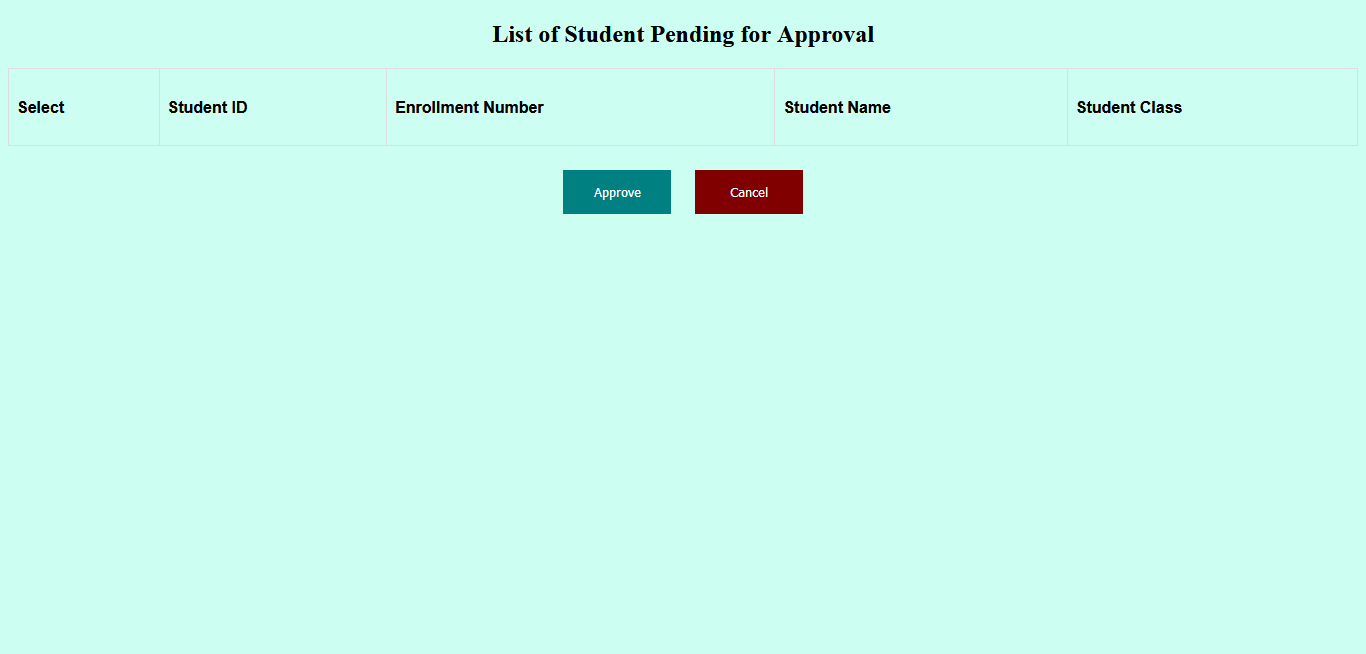
* 1. **Screenshots of Working of Project :-**
     1. **Home Page:**

****

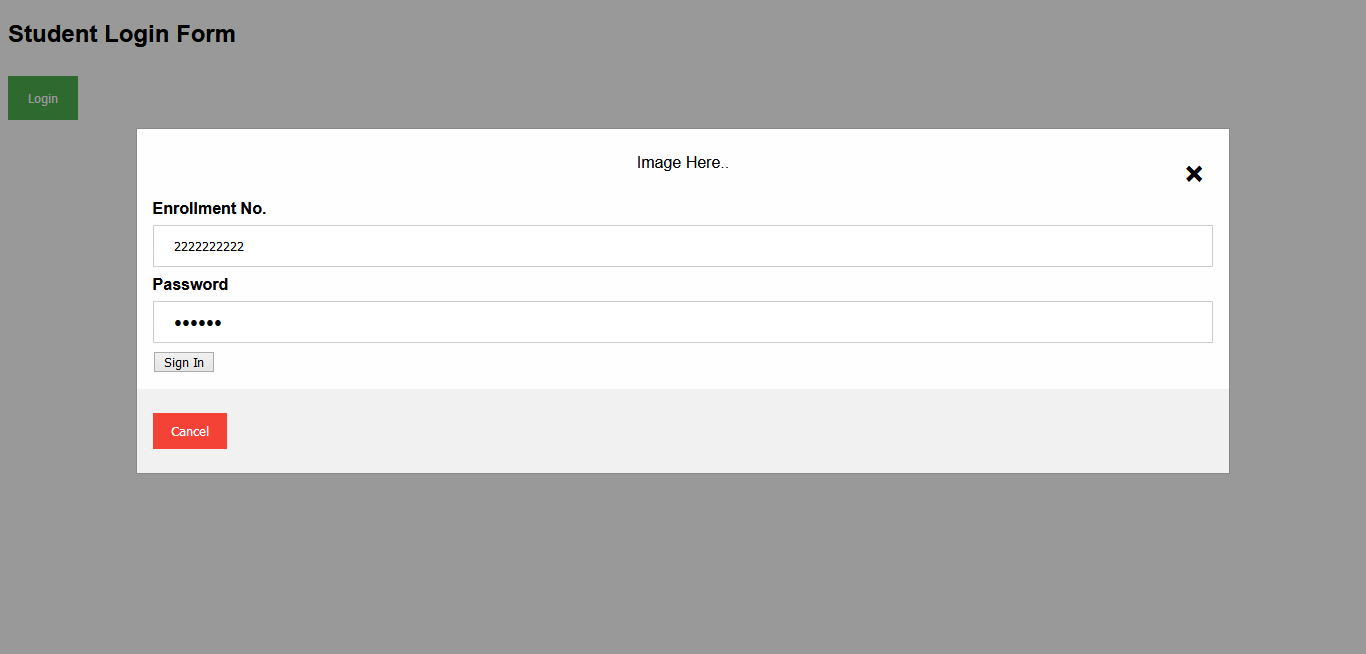
* + 1. **Student Registration Form:**

****

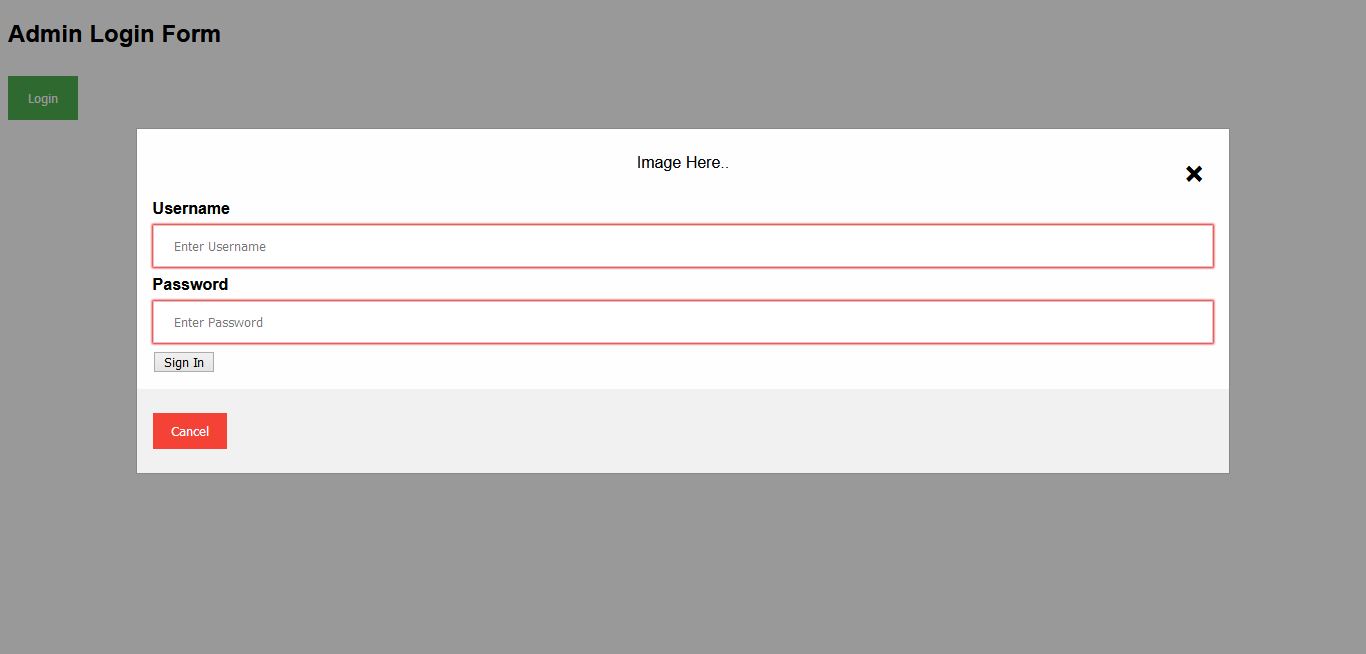
* + 1. **Student Approval:**

****

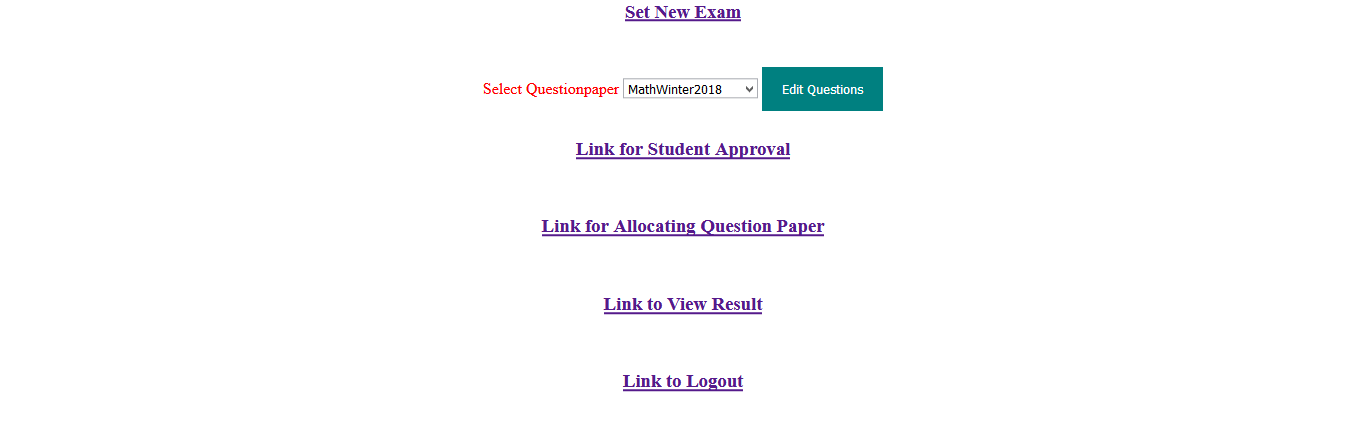
* + 1. **Student Login:**

****

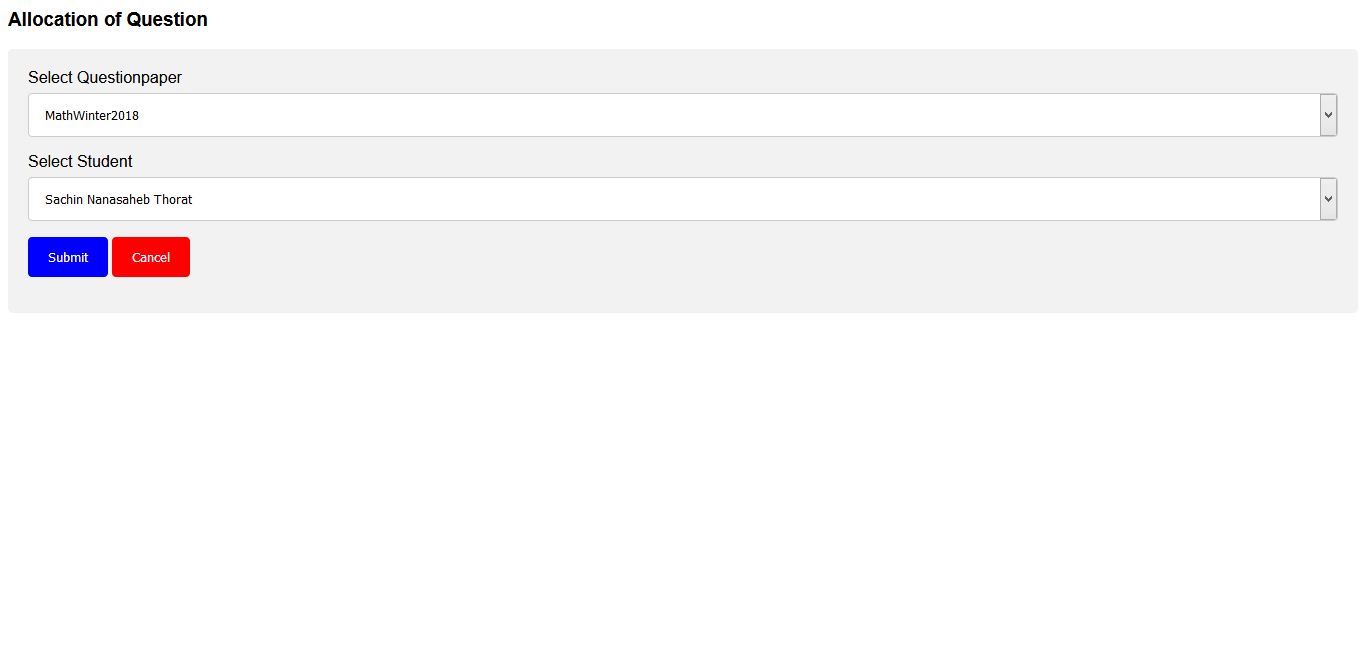
* + 1. **Admin Login:**

****

* + 1. **Admin Dashboard:**

****

* + 1. **Check Result:**

****

**6.1.8 Actual Examination:-**



**CHAPTER 8**

**TESTING**

**8.1 Introduction:**

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test , with respect to the context in which it is intended to operate. This includes, but is not limited to, the process of executing a program or application with the intent of finding software bugs. It can also be stated as the process of validating and verifying that a software program/application/product meets the Cabiness and technical requirements that guided its design and development, so that it works as expected and can be implemented with the same characteristics. A primary purpose for testing is to detect software failures so that defects may be uncovered and corrected. This is a non-trivial pursuit. Testing cannot establish that a product functions properly under all conditions but can only establish that it does not function properly under specific conditions. The scope of software testing often includes examination of code as well as execution of that code in various environments and conditions as well as examining the aspects of code: does it do what it is supposed to do and do what it needs to do. In the current culture of software development, a testing organization may be separate from the development team. There are various roles for testing team members. Information derived from software testing may be used to correct the process by which software is developed.

**8.1.1 Defects and failures**

Not all software defects are caused by coding errors. One common source of expensive defects is caused by requirements gaps, e.g., unrecognized requirements, that result in errors of omission by the program designer. A common source of requirements gaps is non-functional requirements such as testability, scalability, maintainability, usability, performance, and security.Software faults occur through the following process. A programmer makes an error (mistake), which results in a defect (fault, bug) in the software source code. If this defect is executed, in certain situations the system will produce wrong results, causing a failure.Not all defects will necessarily result in failures. For example,defects in dead code will never result in failures. A defect can turn into a failure when the environment is changed. Examples of these changes in environment include thesoftware being run on a new hardware platform, alterations in source data or interacting with different software. A single defect may result in a wide range of failure symptoms.

**8.1.2 Compatibility**

A frequent cause of software failure is compatibility with another application, a new operating system, or, increasingly, web browser version. In the case of lack of backward compatibility, this can occur (for example...) because the programmers have only considered coding their programs for, or testing the software upon, "the latest version of" this-or-that operating system. The unintended consequence of this fact is that: their latest work might not be fully compatible with earlier mixtures of software/hardware, or it might not be fully compatible with another important operating system. In any case, these differences, whatever they might be, may have resulted in (unintended...) software failures, as witnessed by some significant population of computer users.

This could be considered a "prevention oriented strategy" that fits well with the latest testing phase suggested by Dave Gelperin and William C. Hetzel, as cited below.

**8.1.3 Input combinations and preconditions**

A very fundamental problem with software testing is that testing under all combinations of inputs and preconditions (initial state) is not feasible, even with a simple product. This means that the number of defects in a software product can be very large and defects that occur infrequently are difficult to find in testing. More significantly, non-functional dimensions of quality (how it is supposed to be versus what it is supposed to do) -- for example, usability, scalability, performance, compatibility, reliability -- can be highly subjective; something that constitutes sufficient value to one person may be intolerable to another.

**8.1.4 Static vs. dynamic testing**

There are many approaches to software testing. Reviews, walkthroughs or inspections are considered as static testing, whereas actually executing programmed code with a given set of test cases is referred to as dynamic testing. The former can be, (and unfortunately in practice often is...) omitted, whereas the latter takes placewhen programs begin to be used for the first time - which is normally considered the beginning of the testing stage. This may actually begin before the program is 100% complete in order to test particular sections of code (modules or discrete functions).

For example, Spreadsheet programs are, by their very nature, tested to a large extent "on the fly" during the build process as the result of some calculation or text manipulation is shown interactively immediately after each formula is entered.

**8.2 Types of Testing:-**

**8.2.1. Unit Testing:**

This is the smallest testable unit of a computer system and is normally tested using the white box testing. The author of the programs usually carries out unit tests. We have performed the unit testing of our Project by taking each Unit Individually.

* + 1. **Integration Testing:**

In integration testing, the different units of the system are integrated together to form the complete system and this type of testing checks the system as whole to ensure that it is doing what is supposed to do. The testing of an integrated system can be carried out top-down, bottom-up, or big-bang. In this type of testing, some parts will be tested with white box testing and some with black box testing techniques. This type of testing plays very important role in increasing the systems productivity. We have checked our system by using the integration testing techniques. We have tested it by allowing more than one Student to appear for Exam and Check the resultant System.

* + 1. **System Testing:**

A part from testing the system to validate the functionality of software against the requirements, it is also necessary to test the non-functional aspect of the system. Some examples of non-functional tools include tests to check performance, data security, usability/user friendliness, volume, load/stress that we have used in our project to test the various modules. System testing consists of the following steps:

1. Program(s) testing.

2. String testing.

3. System testing.

4. System documentation.

5. User acceptance testing.

**8.3 Test Cases:-**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description** | **Steps** | **Test Data** | **Steps to Perform** | **Expected Result** | **Test Result (Pass/ Fail)** |
| 1a | Student Login | 1 | Enrollment No. | Enter Enrollment No. | Student Login Page is populated | Pass |
| 2 | Password | Enter Password |
| 3 | 10 Digit Numbers | Press Login |
| 1b | Admin Login | 1 | Username | Enter Username | Admin Dashboard Page is populated | Pass |
| 2 | Password | Enter Password |
| 3 |  | Press Login |
| 2. | Create Exam | 1. | Question Paper Tile | Enter Title | New Question paper Created | Pass |
| 2. | Subject Details | Enter Subject |
| 3. | Class Details | Enter Details |
| 4. | Time | Enter Time |
| 5. | Total Questions | Enter Total Number of Questions |
| 6. | Total marks | Enter Total Marks |
| 3. | Edit Questions | 1. | Question Paper | Select Question Paper | Question is Edited | Pass |
| 2. | Question Number | Press on Edit Button to Edit Questions |
| 3. | Question | Enter the Question |
| 4. | Options | Enter all Options |
| 5. | Marks | Enter Marks |
| 4. | Staff Approval | 1. | Student Approval Table | Select the Checkbox for Student Approval | Students are Approved | Pass |
| 2. |  | Press Approve |
| 5. | Check Result | 1. 1. | Exam | Select Exam from List | Result Displayed | Pass |
| 2. | Student | Select Student name from List |

**CHAPTER 9**

**SCOPE OF FUTURE APPLICATION**

It is directly dependent on the lay stone of the project that is we will have to design a system which when the time passes having a better system initially should not become a joke later. The scope of my software is based on the idea of connecting people online The next increment may scope changes suggested by a review of the preceding increment, but once the second increment commences, scope is again frozen temporarily. This approach enable the web application team to work without having to accommodate a continual stream of change but still recognizes the continuous evolution characteristics of most web application. Besides that, the following basic quality in the software always safeguards the future scope of the software.

**Correctness:-**

When a program functions correctly according to their specification that it show the quality of correctness to the definition of correctness the specification of the system that determine unambiguous. Whether a program meets the specification

**Reusability:**

Reusability is possible as and when we required in this application. We can update its next version. Reusable software reduces design, coding and testing cost by amortizing effort over several designs. Reducing the amount of code also simplified understanding, which increases the likelihood that the code is correct. We followed up both types of reusability as sharing of newly written code within a project and reuse of previously written code on new projects.

**Extensibility:**

This application software is extended in ways that its original developers may not expect. The following principles enhance extensibility like hiding data structures, avoiding traversing multiple links or methods, avoiding case statements on object type and distinguishing public and private operations.

**RoCabtness:**

Its method is roCabt and it will not fail even if it receives improper parameters. There are some alert pages and messages are flashed out with some dialogue boxes to warn and inform the end user about the current processes going on. It also interacts with the user by alerting them about invalid parameters.

**Understandability:**

A method is understandable if anyone other than the developer of the method can understand the code (as well as the developer after a time-span).

**Cost-effectiveness:**

Its cost is under the budget and developed within given time period. It is always desirable to aim for a system with a minimum cost subject to the condition that it must satisfy all the requirements.

**CONCLUSIONS**

The project "Online Examination System” is based on managing the Online Examination of Institutes. This system allow the Admin to maintain all needed details regarding Student Question Paper, Approval and Actual Examination and result of Student. The system provides a graphical user interface, which helps all the users to know the Questions and Submission working very well. This system provides work status report for staff members who work with this system.

This system gives advantage by providing all information on a single click. If any Students want to know about his Result status than this information is also provided by this system. It also generates reports which give detailed information about the Result status and Student status according to this system.

**References**

**11.1 Book Referred:**

1. Java:The Complete Reference .by Steve Suehring, Tim Converse, and Joyce Park.

2. Core Servlets and JavaServer Pages by Marty Hall, Larry Brown.

3. [Effective Java](https://www.amazon.com/Effective-Java-3rd-Joshua-Bloch/dp/0134685997/?tag=javamysqlanta-20) by Joshua Bloch.

4**. Head First Java by O Reilly.**

**11.2 Site Referred:**

http://www.w3schools.com.

<http://www.tutorialpoint.com>.

<http://www.youtube.com>.