A STUDY ON THE TESTING AND ANALYSIS OF PRODUCTION PLANNING AND CONTROL SYSTEM IN ACCESSUS TECHNOLOGIES, CHENNAI

by

D.SUJITHA

(REG.NO.71206631057)

of

Department of Management Studies

Kumaraguru College of Technology

COIMBATORE.

A PROJECT REPORT

submitted to the

FACULTY OF MANAGEMENT STUDIES

in the partial fulfillment of the requirements

for the award of the degree

of

MASTER OF BUSINESS ADMINISTRATION

MAY 2008

KCT Business School
Department of management studies
Kumaraguru College of Technology
Coimbatore – 641006

BONAFIDE CERTIFICATE

Certified that this project titled, **"A STUDY ON THE TESTING AND ANALYSIS OF PRODUCTION PLANNING AND CONTROL SYSTEM OF ACCESSUS TECHNOLOGIES, CHENNAI"** is the bonafide work of SUJITHA.D (Reg no: 71206631057), who carried this research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Project Guide

Director

Evaluated and Viva Voce conducted on 3-7-2008

INTERNAL EXAMINER

3/7/2008
EXTERNAL EXAMINER

DECLARATION

I, hereby declare that this project entitled **"A STUDY ON THE TESTING AND ANALYSIS OF PRODUCTION PLANNING AND CONTROL SYSTEM OF ACCESSUS TECHNOLOGIES, CHENNAI"**, has been undertaken for academic purpose submitted to Anna University, Chennai in partial fulfillment of the requirements for the award of the degree of Master of Business Administration. The project report is the record of the original work done by me under the guidance of Mr.S.Mohanavel, Senior Lecturer during the academic year 2007 – 2008.

I, also declare hereby, that the information given in this report is correct to best of my knowledge and belief.

Date: 3-7-2008

Place: Coimbatore

SUJITHA.D

AccessUs
Technologies

2nd May 2008

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Ms.**D.Sujitha,** (Reg No: 71206631057), final year **M.B.A** student of KCT Business School, Kumaraguru College Of Tech, had undergone a Project titled, **"A STUDY ON THE TESTING AND ANALYSIS OF PRODUCTION PLANNING AND CONTOL SYSTEM IN ACCESSUS TECHNOLOGIES, CHENNAI"**, during the period January 19,2008 to April 14,2008.

During the project work she was sincere and committed. She remained an excellent team player and was totally dedicated throughout the project period.

We wish her all the best for all her future endeavour.

From AccessUs Technologies.

Name and designation
of the organization guide:

Signature of the
Organization guide

Director

Ceebros Arcade, III Floor, No. 1, 3rd Cross Main Road, Kasthurbai Nagar, Adyar, Chennai - 600 020. India.
Phone 91 44 6450 3366 Web : www.accessus.in

ACKNOWLEDGEMENT

I highly wish to express my profound gratitude to **Dr. Joseph. V. Thanikal**, Principal, Kumaraguru College of Technology for providing this opportunity and facilities to carry out this project.

I convey my heartiest thanks to **Dr. S.V.Devanathan,** Director, KCT Business School for permitting me to do the project.

I owe my reverential gratitude to my faculty guide **Mr.S.Mohanavel,** Senior Lecturer, KCT Business School, for her personal ideas and constant advice regarding the project.

I convey my heart felt thanks to **Mr.R. Bhaskaran**, Test Engineer, Accessus Technologies, Chennai and all the staff members for providing me necessary information for the successful completion of this project.

I also thank my parents and friends for their strong support and inspiration provide in bringing out this volume.

EXECUTIVE SUMMARY

.The project entitled "A STUDY ON THE TESTING AND ANALYSIS OF PRODUCTION PLANNING AND CONTROL SYSTEM IN ACCESSUS TECHNOLOGIES, CHENNAI", was undertaken in order to generate test cases, to perform unit test, integration test and system test and analyze the test reports like Test Execution Report, Peer Review Checklist, Defect tracking Report and Test Summary report.

Accessus Technologies, Chennai offers a broad spectrum of IT services that include custom application development, application management, application reengineering and independent testing service. Accessus customers include Fortune 500 companies, multinationals and successful small & medium enterprises across several industries.

In this project Descriptive research design is used.The data collected in this research work is primary data.The following tools are used in the study MS Word, MS excel, QTP(Quick Test professional).

In Production planning and control system 239 test cases were generated out of which 221 were passed and remaining 18 test cases were failed. On the whole Production planning and control system consists of 7.5 % defects which do not fall into the acceptable limit of 3%. The regression testing was done using the automated testing tool named QTP revealed that the other features of the project remained unaltered after the inclusion of a new functionality. The overall satisfaction level of the respondents regarding the individual aspects of the testing process was found to be 75%.

In order to minimize the defect rate, the management should conduct several training programmes and seminars in order to improve the programming skills of the developer's. Both the testers and the developers should be trained effectively to possess both testing and developing skills in order to overcome minor defects by themselves without depending on others.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF CHARTS

1. INTRODUCTION

CHAPTER 1
INTRODUCTION

1.1 BACKGROUND OF STUDY

Software testing is the process used to assess the quality of computer software. Software testing is an empirical technical 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. Quality is not an absolute; it is value to some person. With that in mind, testing can never completely establish the correctness of arbitrary computer software; testing furnishes a criticism or comparison that compares the state and behaviour of the product against a specification. An important point is that software testing should be distinguished from the separate discipline of Software Quality Assurance, which encompasses all business process areas, not just testing. Over its existence, computer software has continued to grow in complexity and size. Every software product has a target audience. For example: video game software has its audience completely different from banking software. Therefore, when an organization develops or otherwise invests in a software product, it presumably must assess whether the software product will be acceptable to its end users, its target audience, its purchasers, and other stakeholders. Software testing is the process of attempting to make this assessment.

Software testing may be viewed as an important part of the software quality assurance process. In SQA, software process specialists and auditors take a broader view on software and its development. They examine and change the software engineering process itself to reduce the amount of faults that end up in defect rate. What constitutes an acceptable defect rate depends on the nature of the software. An arcade video game designed to simulate flying an airplane would presumably have a much higher tolerance for defects than software used to control an actual airliner. Although there are close links with SQA testing departments often exist independently, and there may be no SQA areas in some companies.

The 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 a 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 the software being run on a new hardware platform, alterations in source_data or interacting with different software.

A problem with software testing is that testing all combinations of inputs and preconditions is not feasible when testing anything other than 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, parafunctional dimensions of quality--for example, usability, scalability, performance, compatibility, reliability--can be highly subjective; something that constitutes sufficient value to one person may be intolerable to another.

1.1.1 APPROACHES TO SOFTWARE TESTING:

VERIFICATION

Verification refers to the set of activties that ensure software correctly implements a specific function.

Have we built the software right (i.e., does it match the specification)

VALIDATION

Validation refers to a different set of activities that ensure that software that has been built is traceable to customer requirements.

Have we built the right software (i.e., is this what the customer wants)

1.1.2 HISTORY

The separation of debugging from testing was initially introduced by Glenford J. Myers in 1979. Although his attention was on breakage testing, it illustrated the desire of the software engineering community to separate fundamental development activities, such as debugging, from that of verification. Dr. Dave Gelperin and Dr. William C. Hetzel classified in 1988 the phases and goals in software testing in the following stages:

Until 1956 - Debugging oriented

1957-1978 - Demonstration oriented

1979-1982 - Destruction oriented

1983-1987 - Evaluation oriented

1988-2000 - Prevention oriented

TESTING METHODS

Software testing methods are traditionally divided into

- Black box testing
- White box testing.

Black box testing treats the software as a black-box without any understanding of internal behavior. It aims to test the functionality according to the requirements. Thus, the tester inputs data and only sees the output from the test object. This level of testing usually requires thorough test cases to be provided to the tester who then can simply verify that for a given input, the output value (or behavior), is the same as the expected value specified in the test case. Black box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix etc.

White box testing, however, is when the tester has access to the internal data structures, code, and algorithms. White box testing methods include creating tests to satisfy some code coverage criteria. For example, the test designer can create tests to cause all statements in the program to be executed at least once. Other examples of white box testing are mutation testing and fault injection methods. White box testing includes all static testing.

In recent years the term **Grey box testing** has come into common usage. This involves having access to internal data structures and algorithms for purposes of designing the test cases, but testing at the user, or black-box level. Manipulating input data and formatting output do not qualify as grey-box because the input and output are clearly outside of the black-box we are calling the software under test. This is particularly important when conducting integration testing between two modules of code written by two different developers, where only the interfaces are exposed for test. Grey box testing may also include reverse engineering to determine, for instance, boundary values.

Special methods exist to test non-functional aspects of software. Performance testing checks to see if the software can handle large quantities of data or users. Usability testing is needed to check if the user interface is easy to use and understand. Security testing is essential for software which processes confidential data and to prevent system intrusion by hackers. To test internationalization and localization aspects of software a pseudo localization method can be used.

1.1.3 TESTING PROCESS

A common practice of software testing is performed by an independent group of testers after the functionality is developed before it is shipped to the customer. This practice often results in the testing phase being used as project buffer to compensate for project delays, thereby compromising the time devoted to testing. Another practice is to start software testing at the same moment the project starts and it is a continuous process until the project finishes.

In counterpoint, some emerging software disciplines such as extreme programming and the agile software development movement, adhere to a "test-driven software development" model. In this process unit tests are written first, by the software engineers (often with pair programming in the extreme programming methodology). Of course these tests fail initially; as they are expected to. Then as code is written it passes incrementally larger portions of the test suites. The test suites are continuously updated as new failure conditions and corner cases are discovered, and they are integrated with any regression tests that are developed.

1.1.4 LEVELS OF TESTING

1.1.4.1 UNIT TESTING

Definition

Unit testing tests the minimal software component, or module. Each unit (basic component) of the software is tested to verify that the detailed design for the unit has been correctly implemented. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.

Unit testing is performed by developers.The testing of a single program, module, or unit of code. Validates the software performs as designed. Software unit gets ready for testing with other system component, such as other software units, hardware, documentation or users.

Limitations of unit testing

- Testing, in general, cannot be expected to catch every error in the program. The same is true for unit testing. By definition, it only tests the functionality of the units themselves.
- Unit testing may not catch integration errors, performance problems, or other system-wide issues. Unit testing is more effective if it is used in conjunction with other software testing activities.
- Like all forms of software testing, unit tests can only show the presence of errors; it cannot show the absence of errors.
- To obtain the intended benefits from unit testing, a rigorous sense of discipline is needed throughout the software development process.

1.1.4.2 INTERGRATION TESTING

Definition

Integration testing exposes defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system. It is usually performed by the developers.The testing of a single program, module, or unit of code Validates that multiple parts of the system interact according to the system design. Portion of the system ready for testing with other portion of the system

Limitations

Any conditions not stated in specified integration tests, outside of the confirmation of the execution of design items, will generally not be tested.

1.1.4.3 SYSTEM TESTING

Definition

System testing of software or hardware is testing conducted on a complete; Integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

It is usually performed by both developers and users.The testing of an entire computing system. This kind of testing can include functional and structural testing, such as stress testing. It validates the system requirements. A tested computer system is based on what was specified to be developed or purchased.

Before shipping the final version of software, alpha and beta testing are often done additionally

Alpha testing is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

Beta testing comes after alpha testing. Versions of the software, known as beta versions, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the product has few faults or bugs. Sometimes, beta versions are made available to the open public to increase the feedback field to a maximal number of future users.

1.1.4.4 USER ACCEPTANCE TESTING

Definition

Acceptance testing can be conducted by the end-user, customer, or client to validate whether or not to accept the product. Acceptance testing may be performed as part of the hand-off process between any two phases of development.

It is performed by users.The testing of computer system or parts of the computer system to make sure it will work in the system regardless of what the system requirement indicate.A tested computer system, based on user needs

1.1.4.5 REGRESSION TESTING

After modifying software, either for a change in functionality or to fix defects, regression re-runs previously passing tests on the modified software to ensure that the modifications haven't unintentionally caused a regression of previous functionality. Regression testing can be performed at any or all of the above test levels. These regression tests are often automated. More specific forms of regression testing are known as sanity testing, when quickly checking for bizarre behaviour, and smoke testing when testing for basic functionality.

1.2 REVIEW OF LITERATURE

Abdeslam En-Nouaary, "Test Selection Criteria for Real-time Systems Modeled as timed Input-output Automata," International Journal of Web Information Systems; Volume: 3 Issue: 4; 2007. This paper aims to address formal testing of real-time systems by providing readers with guidance for generating test cases from timed automata. Design/methodology/approach – In this paper, a set of test selection criteria is presented. Such criteria are useful for testing real-time systems specified by timed automata. The criteria are introduced after the presentation of timed automata model and the concepts related to it. Findings – The paper finds that the set of test selection criteria are ordered based on the inclusion relation. The ordering is useful for developing new testing methods and for comparing existing approaches. Originality/value – Each of the proposed test selection criteria can be used to develop a new method for testing timed automata with certain fault coverage.

David Talby, Orit Hazzan, Yael Dubinsky, Arie Keren, "Agile Software Testing in a Large-Scale Project," IEEE Software, vol. 23, no. 4, pp. 30-37, Jul/Aug, 2006. Agile software development implies major changes in the way you test software, from strategic project planning to day-to-day activities. While previous publications focus on programmer-oriented practices, such as test-first programming, they've neglected many issues related to large-scale projects and to an organization's initial transition to agile methods. This article attempts to close this gap, reporting on the Israeli Air Force's Extreme Programming experiences in a large-scale, enterprise-critical software project. Along with qualitative and quantitative data that validates agile testing practices, the article presents guidelines for test design and execution, working with professional testers, planning the quality process, and defect management. This article is part of a special issue on Software Testing.

Gerard Meszaros, Janice Aston, "Adding Usability Testing to an Agile Project," agile, pp. 289-294, AGILE 2006 (AGILE'06), 2006. Usability testing based on paper prototypes and early versions of the software were added to the agile development process for the second application release resulting in a significant reduction of usability related rework. The paper prototype became a tangible representation of the project vision that was used in many ways that contributed to the resounding success of the project.

Jamie Dobson, "Performance Testing on an Agile Project," agile, pp. 351-358, AGILE 2007 (AGILE 2007), 2007. This experience report is about a software process, designed with performance testing in mind, and was used to create a mission critical integration layer. This report focuses on how the team carried out performance tests iteratively, how a large array of processes and techniques were employed to support a specialist performance test team, and how changes in management style were needed to support performance testing on an agile project.

Jason McDonald, Leesa Murray, Peter Lindsay, Paul Strooper, "Module Testing Embedded Software--An Industrial Pilot Project," 0233, Seventh IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'01), 2001. This paper reports on an industrial pilot project that introduces systematic, automated module testing for embedded software in distributed, real-time, control systems. The systems are used in safety-related applications, are complex in nature, and hence have strong requirements for test coverage, audit ability and repeatability. This paper explores issues of isolating modules from the run-time environment, improving integration of testing into the development environment, automating testing, and improving test planning and documentation. Metrics were gathered throughout the project that allows a coarse cost-benefit evaluation. Code coverage metrics for statement and branch coverage were also gathered using a commercial code coverage analysis tool. The testing exposed a number of latent faults within the software, and the overall results of the project show that module testing is feasible for this complex, embedded software.

Maaret Pyhajarvi, Kristian Rautiainen, Juha Itkonen, "Increasing Understanding of the Modern Testing Perspective in Software Product Development Projects," p. 250b, 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 8, 2003. Testing can be difficult to integrate into software development. Approaches to software testing in relation to implementing software are based on the V-model of testing. The software process behind the V-model is the traditional waterfall model, and as such the traditional testing approaches cannot take iterative, incremental and agile approaches to developing software into account well enough. In this paper, we describe the use of a general iterative and incremental framework defined for controlling product development — 4CC — from a modern testing perspective. The framework provides a common language in which the implementation details and pacing as well as testing details and pacing in software product development projects can be communicated. Viewing testing through a general iterative and incremental framework adds to understanding how the testing process should be defined and improved in relation to the software development process. Additionally, best practices for testing are identified.

Robert Pavur, Maliyakal Jayakumar, Howard Clayton, " Software testing metrics: do they have merit?," Industrial Management & Data Systems; Volume: 99 Issue: 1; 1999 Technical paper. Project managers in information systems play a central role in the development, maintenance, and enhancement of software. Software metrics assist these managers in identifying opportunities for process improvement and help quantify software characteristics. Weaknesses in the traditional approaches to measuring reliability have led to the development of software metrics. The interpretation of software metrics can be critical to making effective responses in the management information systems' decision-making processes. This paper gives insight into the use and understanding of some software metrics.

Tom Hankinson, "Software Testing Goes Underground," Industrial Management & Data Systems; Volume: 93 Issue: 4; 1993. At London Underground Ltd (LUL), getting engineers and their equipment to and from site quickly and safely is a major task for its engineering safety support division. A huge DEC VAX database helps the division maximize the time available for track maintenance and repair. The database is the core of LUL's Co-ordination of Railway Engineering Works (CREW) computer system. The introduction of automated software testing has improved the quality of the system as enhancements are introduced. The story reveals how manual testing had not only proved a bottleneck in development but was incapable of delivering the required level of consistency; neither could it properly assess CREW's performance under heavy load.

Tom Hankinson, "A testing time for software," The TQM Journal; Volume: 3 Issue: 4; 1991 suggests, with statistics, that a substantial part of any software development budget is spent on revisions and maintenance. Argues that systems are long overdue, to measure the quality of software applications rather than focusing on the quality of the procedures and management approach. Feels the quest for software quality must not be held up by standards debates or management philosophies and that investment in CASE tools or 4GLs, though valuable, simply puts the majority of software budgets into areas where the smallest productivity improvements are to be made. Argues that as only a tiny percentage of the money spent on software tools is devoted to products for application testing, yet test and maintenance consume over half the development effort, it is time this imbalance was rectified.

Yash P. Gupta, "Software Quality Assurance," International Journal of Quality & Reliability Management; Volume: 6 Issue: 4;
In this article the following are discussed: (a) the role of quality assurance in the development of computer software systems; (b) factors of a successful quality assurance programme; and (c) the steps required in the development of such a programme. The implications for quality assurance for management are also considered.

Zhonglin He, Geoff Staples, Margaret Ross, Ian Court, Keith Hazzard, " Orthogonal software testing: Taguchi methods in software unit and subsystem testing," Logistics Information Management; Volume: 10 Issue: 5; 1997 Technical paper Suggests that, in order to detect and correct software defects as early as possible, identifying and generating more defect-sensitive test cases for software unit and subsystem testing is one solution. Proposes an orthogonal software testing approach, based on the quality optimization techniques namely Taguchi methods. This orthogonal approach treats the input parameters of a software unit or subsystem as design factors in an orthogonal array, and stratifies input parameter domains into equivalent classes to form levels of factors. Describes how test cases are generated statistically for each trial of factorial orthogonal experiments. The adequacy of the generated test cases can be validated by examining testing coverage metrics. The results of test case executions can be analyzed in order to find the sensibility of test cases for detecting defects, to generate more effective test cases in further testing, and to help locate and correct defects in the early stage of testing.

1.3 STATEMENT OF THE PROBLEM

OVERVIEW OF THE PROJECT:

Introduction

Production planning and control system is designed and developed according to the user requirements. It may be better choice to select SQL-Server to maintain the database and Visual Basic .Net for front end programming. Tables created will have less chance for corruption, when compared to other databases. The .Net Technologies enable the application with scalability and extended performance measurements for the industrial management.

Production planning and control system was developed for

- Instantaneous retrieval of any type of information.
- Generation of Query based reports.
- To provide administrator for maintenance of all the master files.
- Provide administrator to generate, control and analyze reports.

Production planning control system contains the following modules:

1. Job Order System
2. Productivity Evaluation
3. Production Process

JOB ORDER SYSTEM

This module contains the following sub-modules:

1. Work order register

Every work in the factory is done against a written order known as "work order ".Information of all such work orders is collected in a register known as "work order register". After completion of work, entries are made in this register for future references and keeping permanent record.

2. Job Card

This module maintained the details of (After receiving the work order, foreman splits up) the job in different parts according to different operation required on it. The work is assigned to the workers on a card known as job card.

3. Route Card

This module presents the material used and their progress from machine operation to machine operation. The job specifications are essential for the foreman and others to know exactly the nature of the job to be done.

PRODUCTIVITY EVALUATION

This module maintains the details of the productivity. Productivity is mainly used to reduce the wastage of resources. The resources may be men, materials, machines ,power, space, time, building etc., It is a comparison between the quantity of goods and services produced (output) and the quantity of resources used to produce these goods and services(Inputs).

This module contains the following sub-module:

1. General Productivity

This sub-module calculates the ratio between output and input.

2. Labour Productivity

The productivity of labour can be increased by increasing efficiency of labour and reducing idle time.

a. In Terms Of Hours

This module calculates the ratio between production in standard hours and actual man hours.

b. In Terms Of Money

This module evaluates ratio between total cost of output and no of workers.

3. Material Productivity

This sub-module determines the ratio between material cost and number of units produced.

This productivity can be increased by using minimum material with the help of skilled workers, adequate machine tools and good design of product.

4. Machine Productivity

This sub-module evaluates the ratio between output in standard hours and actual machine hours. That is to know the machine capacity.

PRODUCTION PROCESS

This module contains the following sub-modules

1. Item Master specifies the details of the item to be produced.
2. Customer Master may include the details of the customer.
3. Machine detail generates the details of the machines used to produce.
4. Production Detail may contain the details of the production process.
5. Rejection Detail specifies the details of the item to be rejected.

1.3 OBJECTIVES

1.4.1 PRIMARY OBJECTIVE

To prepare test plan, generate test cases, perform unit testing, system testing, integration testing and to study and analyze the various test reports on Human resource procurement system of Accessus Technologies, Chennai.

1.4.2 SECONDARY OBJECTIVE

- To demonstrate that software functions appear to be working according to specification, those performance requirements appear to have been met.
- To demonstrate that testing provides good indication of software reliability and quality.

1.5 SCOPE OF THE STUDY

The main aim of our project is to design different test cases that systematically uncover different classes of errors and also with minimum amount of time and effort.Testing is used to determine the status of the product during and after the process. Testing provides good indication of software reliability and quality. The purpose of our project is to test and analyze the various test reports of Human Resource procurement System. In this project Unit Testing, Integration testing and System is performed in order to find out the bugs.

1.6 RESEARCH METHODOLOGY

1.6.1 TYPE OF STUDY:

The primary research is descriptive in nature. Descriptive research is appropriate since it helps in identifying the new opportunities present in the decision situation and it also serves as a step towards further research activity. The study is aimed at Testing and analysis of the software project

1.6.2 METHOD OF DATA COLLECTION:

The data used for this study is primary in nature.(Test cases, peer review check list,defect tracking report)

1.6.3 TOOLS FOR ANALYSIS:

- MS Excel

  Using MS Excel the collected data were analyzed and tabulated.

- QTP- Quick test professional

  An automated test tool Quick Test Professional was used for Regression testing. Quick test enables to test standard Web objects, ActiveX controls, and Visual Basic control. Manual testing is time consuming and tedious, requiring a heavy investments in human resources. Automated testing with Quick Test addresses these problems dramatically speeding up the testing process. The Quick Test testing process consists of 7 main phase:

  - Preparing to record
  - Recording a session on the application
  - Enhancing the test
  - Debugging the testRunning the test
  - Analyzing the test
  - Reporting the defects

1.6.4 TEST PLAN

Revision History

DATE	VERSION	AUTHOR	DESCRIPTION
02/15/2008	1.0	Sujitha.D	Initial draft
02/20/2008	1.1	Sujitha.D	Second draft
02/25/2008	1.2	Sujitha.D	Final draft

1.6.4.1 INTRODUCTION

This software allows Production Planning and control department to achieve their target. Preplanning will increase production at a determined period of time, which will bring out well-planned quality production.

The new Production Planning and Control system will do the following:

- Instantaneous retrieval of any type of information.
- Generation of Query based reports.
- Provides administrator for maintenance of all the master files.
- Provides administrator to generate, control and analyze reports

Test plan objectives

This Test Plan for the new Production Planning and control supports the following objectives:

- Define the activities required to prepare for and conduct System, Beta and User Acceptance testing.
- Communicate to all responsible parties the System Test strategy, dependencies and risks.
- Define deliverables and responsible parties.

1.6.4.2 SCOPE

1.6.4.2.1 Job Order System

Work order register

It should provide work order system in which Information of all work orders are collected in a register known as "work order register". After completion of work, entries are made in this register for future references and keeping permanent record.

Job Card and Route Card

Job card will maintain the details of the job in different parts according to different operation required on it. The Job card represents work is assigned to the workers. Route Card presents the material used and their progress from machine operation to machine operation.

1.6.4.2.2 Productivity Evaluation

Productivity Evaluation module should maintain the details of the productivity by comparing the quantity of goods and services produced (output) and the quantity of resources used to produce these goods and services (Inputs).It evaluates all productivity details such as Labour Productivity, Material Productivity, and Machine Productivity.

1.6.4.2.3. Productivity Process

It should specifies the details of customers, details of item to be produced, machines used to produce, its production process details and the details of items to be rejected.

1.6.4.3. TEST STRATEGY

The test strategy consists of a series of different tests that will fully exercise the Production Planning and Control. The primary purpose of these tests is to uncover the systems limitations and measure its full capabilities. A list of the various planned tests and a brief explanation follows below.

Functionality Test

Functionality test will ensure the functional behavior of Production Planning and Control application. Here the validation of actual functionality of controls are checked after giving input to certain fields in the application

System Test

The System tests will focus on the behavior of the Production Planning and Control. User scenarios will be executed against the system as well as screen mapping and error message testing. Overall, the system tests will test the integrated system and verify that it meets the requirements defined in the requirements document.

Security Test

Security tests will determine how secure the new Production Planning and Control is. The tests will verify that unauthorized user access to confidential data is prevented.

Recovery Test

Recovery tests will force the system to fail in a various ways and verify the recovery is properly performed. It is vitally important that all payroll data is recovered after a system failure & no corruption of the data occurred.

Documentation Test

Tests will be conducted to check the accuracy of the user documentation. These tests will ensure that no features are missing, and the contents can be easily understood.

Beta Test

The Client will beta tests the Production Planning and control System and will report any defects they find. This will subject the system to tests that could not be performed in our test environment.

Regression Test

Regression Test will be conducted due to the necessary activities performed on modified software to provide confidence that the changes are correct and do not adversely affect other system components. While adding new features to our application, the regression test is conducted to ensure that the added new feature should not impact the existing functionalities in our application.

User Acceptance Test

Once the Production Planning and control System is ready for implementation, the client will perform User Acceptance Testing. The purpose of these tests is to confirm that the system is developed according to the specified user requirements and is ready for operational use.

1.6.4.4 ENVIRONMENT REQUIREMENTS

Data Entry workstations

Processor	-	Pentium IV
Clock Speed	-	900 MHz
RAM capacity	-	256MB
Hard Disk	-	40GB
Monitor Type	-	15 Inch Color Monitor
Mouse Type	-	ps/2

Software Configuration

Front End	-	VB. NET
Back End	-	MS –SQL Server 2000
Operating System	-	Windows Xp
Report Tool	-	Crystal Report

1.6.4.5 TEST SCHEDULE

▪ Ramp up / System familiarization	03/05/2008	-	03/06/2008
▪ System Test	03/07/2008	-	03/12/2008
▪ Beta Test	03/13/2008	-	03/14/2008
▪ User Acceptance Test	03/15/2008	-	03/17/2008

1.6.4.6 CONTROL PROCEDURES

Reviews

The project team will perform reviews for each Phase. (i.e. Requirements Review, Design Review, Code Review, Test Plan Review, Test Case Review and Final Test Summary Review). A meeting notice, with related documents, will be emailed to each participant.

Bug Review meetings

Regular weekly meeting will be held to discuss reported defects. The development department will provide status/updates on all defects reported and the test department will provide addition defect information if needed. All member of the project team will participate.

Change Request

Once testing begins, changes to the payroll system are discouraged. If functional changes are required, these proposed changes will be discussed with the Change Control Board (CCB). The CCB will determine the impact of the change and if/when it should be implemented.

Defect Reporting

When defects are found, the testers will complete a defect report on the defect tracking system. The defect tracking Systems is accessible by testers, developers & all members of the project team. When a defect has been fixed or more information is needed, the developer will change the status of the defect to indicate the current state. Once a defect is verified as FIXED, by the testers, the testers will close the defect report.

1.6.4.7 FUNCTIONS TO BE TESTED

The following is a list of functions that will be tested:

- Login for Existing user and Creation of new user
- Search /Look up information
- Escape to return to Main Menu
- Security features
- Error messages
- Report Printing
- Customer Details
- Product Details
- Job cards and Route cards
- Work order Details
- Production details
- Rejection Details
- Machinery Details
- Crystal report for Work order
- Crystal report for Production
- Crystal report for Rejection
- Crystal report for Machinery
- And calculations involved in the application

1.6.4.8 RESOURCES

The Test Lead and Project Manager will determine when system test will start and end. The Test lead will also be responsible for coordinating schedules, equipment, & tools for the testers as well as writing/updating the Test Plan, Weekly Test Status reports and Final Test Summary report. The testers will be responsible for writing the test cases and executing the tests. With the help of the Senior Test Engineer and Test Lead, Client will be responsible for the Beta and User Acceptance tests.

Resources

The test team will consist of:

- A Project Manager
- A Test Lead
- 5 Testers
- A Team from client
- Production Department Manager

1.6.4.9 DELIVERABLES

Deliverables	Responsibility
Develop Test cases	Testers
Test Case Review	Test Lead, Dev. Lead, Testers
Develop Automated test suites	Testers
Obtain User ids and Passwords for payroll system/database	Test Lead
Execute manual and automated tests	Testers & Test Lead
Complete Defect Reports	Everyone testing the product
Document and communicate test status/coverage	Test Lead
Execute Beta tests	A team from client
Document and communicate Beta test status/coverage	Production Department Manager
Execute User Acceptance tests	A team from client
Document and communicate Acceptance test status/coverage	Production Department Manager
Final Test Summary Report	Test Lead

1.6.4.10 SUSPENSION / EXIT CRITERIA

If any defects are found which seriously impact the test progress, the QA manager may choose to Suspend testing Criteria that will justify test suspension are:

- Hardware/software is not available at the times indicated in the project schedule.
- Source code contains one or more critical defects, which seriously prevents or limits testing progress.
- Assigned test resources are not available when needed by the test team.

1.6.4.11 RESUMPTION CRITERIA

If testing is suspended, resumption will only occur when the problem(s) that caused the suspension has been resolved. When a critical defect is the cause of the suspension, the "FIX" must be verified by the test department before testing is resumed.

1.6.4.12 DEPENDENCIES

Personnel Dependencies

The test team requires experience testers to develop, perform and validate these tests. The test team will also need the following resources available: Application developers and Clients.

Software Dependencies

The source code must be unit tested and provided within the scheduled time outlined in the Project Schedule.

Hardware Dependencies

PCs (with specified hardware/software) need to be available during normal working hours. Any downtime will affect the test schedule.

Test Data & Database

Test data & database should also be made available to the testers for use during testing.

1.6.4.13 RISKS

Schedule

The schedule for each phase is very aggressive and could affect testing. A slip in the schedule in one of the other phases could result in a subsequent slip in the test phase. Close project management is crucial to meeting the forecasted completion date.

Management

Management support is required so when the project falls behind, the test schedule does not get squeezed to make up for the delay. Management can reduce the risk of delays by supporting the test team throughout the testing phase and assigning people to this project with the required skills set.

Personnel

Due to the aggressive schedule, it is very important to have experienced testers on this project. Unexpected turnovers can impact the schedule. If attrition does happen, all efforts must be made to replace the experienced individual

Requirements

The test plan and test schedule are based on the current Requirements Document. Any changes to the requirements could affect the test schedule and will need to be approved by the CCB.

1.6.4.14 TOOLS

The Quick Test Professional test tool will be used to help test the new payroll system. We have the licensed product onsite and installed. All of the testers have been trained on the use of this test tool.

1.6.4.15 DOCUMENTATION

The following documentation will be available at the end of the test phase:

- Test Plan
- Test Cases
- Test Case review
- Defect reports
- Final Test Summary Report

1.6.4.16 APPROVALS

Name (Print)	Signature	Date
1.		
2.		
3.		

1.6.5 PREPARATION OF TEST CASES

Validation checks are performed on the following fields.

Text Field

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data

The above testing is done by taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have they entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system.

Using Artificial Test Data:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

Quality Assurance

Quality assurance consists of the auditing and reporting functions of management. The goal of quality assurance is to provide management with the data necessary to be informed about product quality, thereby gaining insight and confide that product quality is meeting its goals. This is an "umbrella activity that is applied throughout the engineering process.

Software quality assurance encompasses

- Analysis, design, coding and testing methods and tools
- Formal technical reviews that are applied during each software engineering
- Multi tiered testing strategy
- Control of software documentation and the change made to it.
- A procedure to ensure compliance with software development standards.
- Measurement and reporting mechanisms.

Quality Factors

The factors that affect the quality can be categorized into two broad groups:

- Factors that can be directly measured.
- Factors that can be indirectly measured

These factors focus on three important aspects of a software product

- Its operational characteristics
- Its ability to undergo change
- Its adaptability to new environment.

Security Technologies and Policies

The software quality assurance is comprised of a variety of tasks associated with seven major activities.

1. Application of technical methods.
2. Conduct of formal technical reviews
3. Software testing
4. Enforcement of standards
5. Control of change
6. Measurement
7. Record keeping and reporting

1.7 LIMITATIONS

- Load testing, Stress testing, Volume Testing were not carried out due to inadequacy of system Components and Facilities.
- Test cases are constructed only in the testers' point of view.

1.8 CHAPTER SCHEME:

CHAPTER 1

This chapter deals with introduction, objective, scope, research methodology, limitations pertaining to the study are covered under this chapter.

CHAPTER 2

This chapter conveys the history of Accessus technologies. This chapter highlights the origin, development, objectives, product profile, market potential, management and future plans of the company.

CHAPTER 3

This chapter gives the macro and micro analysis of Software Testing industry with respect to Accessus Technologies, Chennai.

CHAPTER 4

This chapter deals depicts the data analysis and interpretation.

CHAPTER 5

This chapter gives summary of findings from the study undertaken, the suggestions given and the conclusion.

CHAPTER 6

This chapter gives a list of the references and sources of study

2. ORGANISATION PROFILE

CHAPTER 2

ORGANISATION PROFILE

2.1 HISTORY OF THE ORGANISATION

Accessus Technologies, Chennai offer a broad spectrum of IT services that includes custom application development, application management, application reengineering and independent testing service.Accessus customers include Fortune 500 companies, multinationals and successful small & medium enterprises across several industries.It provides a wide range of information technology solutions that help customers enhance their overall business performance.

2.2 PRODUCT PROFILE

2.2.1 SAP SERVICES

2.2.1.1 Service Offerings

Accessus provides two categories of enterprise consulting services for SAP: Core Services and Value-Added Services. Core SAP services are offered in mission-critical areas that have an immediate and direct impact on deployment and maintenance of the enterprise system. Accessus also offers Value-Added SAP services that facilitate better management of the SAP system.

Core Services	Value added services
Implementation & Rollout	Program Management Services
Application Support	Archival Solutions
Custom Development	Staffing Solutions
Application Integration	
Upgrade	
Expert Consulting	

2.2.1.2 Business solutions

Packaged application implementation can be very challenging. Accessus's Enterprise Consulting team has a good mix of people and process, which help customers get their desired benefits. Accessus predominantly focuses on SAP Product suite.

MySAP ERP

Businesses operate in a dynamic environment; growth, expansion and restructuring result in organizational and process change. Consequently the ERP systems that run the business must keep pace with the changing needs of the organization. Accessus's SAP team focuses on the optimization and enhancement of these deployed ERP. Accessus offers post-implementation services to extend maintain and support live SAP ERP installations to help customers derive maximum advantage from their ERP application. Post-implementation support and maintenance of the system play a critical role in lowering or raising the TCO of SAP application.

Accessus Technologies Enterprise Consulting Services for MySAP ERP

- Application support to ensure business continuity, and effective management of the SAP environment.
- Upgrade and maintenance services for seamless migration to higher versions
- Custom development services to extend functionality of MySAP ERP
- Application integration services to integrate SAP with third-party applications
- Expert consulting and training services for maximizing benefits
- Archival solutions for efficient data management

MySAP CRM

The focus towards a customer centric enterprise requires a world-class CRM solution. These solutions enable the enterprise to obtain end-to-end process harmonization across the enterprise. MySAP CRM integrates customer processes across the enterprise to deliver effective support to activities such as sales, marketing and services. Accessus Technologies offers a complete range of MySAP CRM services from project management and implementation to training and support.

MySAP SRM

MySAP SRM simplifies and automates procurement by integrating purchase processes end-to-end. It facilitates collaboration with suppliers and offers a solution framework to achieve cost and quality advantages from a unified purchasing platform. Accessus offers an array of services to help customers experience the full benefits of MySAP SRM.

Accessus Technologies helps enterprises to implement the mySAP SRM solution that goes a long way to collaborate with the suppliers in ensuring a long-term optimum procurement solution. During implementation of the SRM solution, Accessus Technologies consultants will help enterprises in adapting best procurement practices and tailoring the SRM application in lieu with the client's requirements. Post-implementation, Accessus technologies also offers sustained support, maintenance, upgrade and all critical post-implementation services.

SAP APO

SAP APO (Advanced Planner and Optimizer), a component of the MySAP Supply Chain Management solution, is used in planning and optimizing of supply chain processes at strategic, tactical, and operational levels. Staffed by experts in SCM and SAP APO, Accessus helps customers through all phases of the implementation from creating the SCM model to integrating SAP APO with MySAP ERP. Accessus Technologies helps customers in configuring the APO as per their requirement and developing reports as required assessing various sourcing options

SAP BW

SAP BW is the information backbone engine for SAP and the MySAP landscape. While traditional ERP reporting provides a gamut of operational reports, SAP BW provides comprehensive analysis that aid strategic decision making. Accessus Technologies works with customers in planning, assessment and implementation of SAP BW. Working on the SAP ASAP methodology, Accessus Technologies offers faster implementation and integration of SAP BW with MySAP ERP system and non-SAP systems.

2.2.2 IT SERVICES

2.2.2.1 APPLICATION DEVELOPMENT

Many business processes are unique to enterprises for which no off-the-shelf solutions are available. Custom applications provide competitive advantage by addressing the gaps in the functionality offered by packaged solutions. As demands for custom applications rise, IT departments work hard to design, integrate, implement, support new applications and simultaneously maintain and upgrade legacy systems Accessus Technologies Custom Application Development services offer you new application development with full lifecycle support.

- You benefit from successful, high-quality applications without the hassles of learning curves and staffing.
- Using a phased approach, our experts outline the solution, define the solution architecture, develop prototypes, build details of the solution, develop the solution perform validation against requirements and rollout the solution.
- Accessus Tecnologies use of right combination of tools, frameworks and reusable code, which enables us to quicken application development in specific environments
- Accessus Technologies custom application development services reduce costs and improve performance help and quality to run enterprises efficiently.

Internet usage is burgeoning, placing higher demands on the reliability and the stability of web solutions. Building and deploying web solutions is just the beginning. The bigger challenge is in maintaining and enhancing them when usage volumes ascend and sophisticated functionalities are added. As usage of web solutions is unimpeded by geography and time zones, it is important for solutions to function efficiently on a round the clock basis. Accessus Technologies have the expertise and experience in delivering large web solutions that demand concurrent transaction oriented communications with customers, partners and suppliers. Our web solution experts have an in-depth knowledge of multi-tier application design, data and process integration, established and emerging web technology standards

2.2.2.2 APPLICATION MANAGEMENT

Over decades, enterprises have made investments in IT assets to carry out their business and realize suitable ROI. The IT assets range from legacy platforms to client-server systems to the present-day multi-tier browser based systems. Business successes largely depend on how effectively IT assets and applications are maintained to serve the ever-changing business needs.

Accessus's application management services features and benefits include:

- Accessus offer corrective maintenance, adaptive maintenance and preventive maintenance for your application portfolio
- Accessus support custom applications as well as commercial products such as SAP, Oracle and PeopleSoft.
- Accessus use of right combination of tools, frameworks and reusable code, which enables us to quicken application development in specific environments
- Accessus deal with all your application management problems such as unstructured code, missing documentation, swelling operating costs, performance deteriorations and maintenance staff morales.
- Customers free up ongoing management resources to invest in additional innovation

2.2.2.3 APPLICATION REENGINEERING

Enterprises are increasingly facing the challenge of ageing and maintenance expensive legacy systems, which they need to transform to the latest technologies that provide greater business flexibility and lowest TCO. In such scenarios, multiple applications with overlapping functionality often exist. There is an immense potential for enabling competitive advantage and cost reduction by consolidation, accomplished by reengineering.

2.2.2.4 APPLICATION TESTING

Software testing has changed vastly in the last two decades. With the proliferation of Internet, accessibility of real-time information has attained priority. The success of enterprises largely depends on the availability, reliability, timeliness, accessibility, ease of use, and security of applications. Application testing from being a negligent, amateurish and low-key activity has become a significant stage in the development lifecycle. Accessus Technologies verification and validation testing services are designed to provide independent application testing services. Accessus Technologies independent testing solutions bring in rigorous verification and validation techniques to reduce defects in every stage of the development cycle. Accessus Technologies expert testing teams understand the need to deal with situations such as reduced project timelines, recurrent application modifications, non-availability of clearly defined business requirements, greater than before security apprehensions and erratic user loads. C independent testing services help you drastically reduce bug fixing and maintenance costs and instead focus on adding new features and enhancements to applications. Accessus Technologies testing services ensure superior delivery assurance, quickest time to market and lowest total cost of ownership to our customers.

2.3 FUTURE PLANS

- To achieve consistency in business-wise growth, and their contributions to top & bottom lines
- To increase productivity and efficiency in all businesses, towards better profitability
- Manage timely and appropriate HR
- Manage Risk
- To focus on jobs with larger scale and integrated scope

3. MACRO-MICRO ANALYSIS

CHAPTER 3
MICRO-MACRO ANALYSIS

Testing accounts for almost 30 percent of the software development market. The global testing market is estimated to be a $13 billion industry. With the boom of the third party software testing business, the need for quality and trained manpower has become a critical issue in the industry.

Software testing has been closely associated with development until a few years ago. Now software testing as a career is evolving rapidly and the profile of a successful tester is remarkably different from that of a successful developer. As testing gains more prominence, a generic career road map for software testing is likely to be established in the years to come.

THE INDIAN TESTING MARKET

The size of the Indian testing market is estimated to be Rs 150-200 crore. It is said to be growing at a faster rate than the global average. The software testing arena in India is estimated to require more than 16,000-18,000 professionals within the next one year. The market for software testers in India is likely to open in a big way due to the following reasons:

- Availability of testers and their use of automated tools.
- Indian software testing companies can offer testing services at a fraction of the cost in most other parts of the world.
- Quality of deliverables.
- Turnaround time for delivery.

DEMAND-SUPPLY SITUATION

The supply is not at par with the demand due to the paucity of professional testers in India and abroad, and lack of awareness about testing as a career.Software testing cannot be considered as an alternative to software development. Testing and development require different profiles.

TRAINING IN SOFTWARE TESTING

Professional training is required to start a career in software testing. Training on the concepts of testing on different hardware configurations, processes, using testing tools and other test enablers, working with test management software, defect tracking tools, etc., are required before working on a live testing project.

Following are some India-based organisations that impart training in software testing:

- Quality Solutions for Information Technology (QSIT): A Bangalore based IT related process consulting and high-end training company with partners in China, and Vietnam. IIST partners with QSIT for certification programmes in software testing-Certified Security Testing Professional (CSTP).
- Vyom SQTL Labs (SQTL): A Pune-based company focused on training, recruitment assistance and consulting in software quality, testing, IT infrastructure and service management.
- Amity Soft, Chennai.
- QAI India.
- Anna University: Offers software testing as one of the subjects in many courses.
- Electronics Test and Development Centre, Chennai.

Market opportunities for the Indian offshore software testing companies in 2005 is estimated at US$2 billion—according to industry estimates, it could exceed US$8 billion by 2008.Stepping into the emerging era of "24x7 enterprises", where business can never sleep, organizations rely on the IT infrastructure 24 hours a day. Each little application becomes critical and no error can be termed small. In this new context, the true value of testing assumes greater significance. According to a recent research titled, "Driving Business Value from IT: Optimizing the New IT Environment," conducted for Mercury by the Economist Intelligence Unit on the top IT challenges and priorities, there is an increasing focus on improving quality of IT solutions.

There are number of cases, where the businesses have had significant financial and credibility losses, due to deployment of untested systems. As per one study, the US industry had incurred losses to the tune of US$60 billion in 2002 due to software defects. The Industry: An assessment by Avendus Advisors places the global market size

for software testing at US$100 billion. The share of the independent testing companies in India was US$20 million in 2003-04.According to some estimates, the market opportunities for the Indian offshore software testing companies in the current year could be US$2 billion. Industries expect the market size to touch US$ 8 billion by 2008.

INDIA MAY BECOME SOFTWARE TESTING HUB

Software bugs or errors cost the US economy an estimated $ 59.5 billion annually, which is equivalent to 0.6 per cent of its gross domestic product. This loss is borne by users and developers of software. At least a third of this cost can be saved through better quality and timely software testing.

In India, the market for IT-related services is estimated to reach the $20-billion mark next year. Reported software testing revenues form 1-2 per cent of this figure. Ideally 20-30 per cent of the software development life-cycle (SDLC) should be dedicated to software testing. In today's business, this will empower companies with huge cost advantages, where cost-control and value-creation are paramount. Independent software testing players spotted this huge potential almost six years ago.

Sensing the opportunities in this emerging space, most of the top ten IT companies in India created independent testing practices in-house. Leading IT companies now position testing as a standalone service and often find it an easier route to gain entry into new accounts for offshoring services. Revenues from testing services are poised to grow exponentially over the next five years. The following trends are fuelling this growth:

Worldwide, CIOs (chief information officers) are under pressure to increase the ROI (return-on-investment) on their IT investments. At the organizational level, most institutions in the developed world have realized that their survival depend on how effectively they can cut operational costs while improving the service standards to their constituents. IT outsourcing, and subsequently, off shoring have been the answer to these challenges. One out of every four companies is now outsourcing an IT function. In 2003, Europe witnessed an 87 per cent jump in outsourcing. Independent testing from an early stage in the SDLC is seen as an effective way to mitigate the risks of outsourcing.

Project time-lines and budgets are crashing. By the time projects come to the testing stage, over 50 per cent would be staring at unreasonable deadlines to go live. Users and providers of technology recognize that independent testing companies can bring down testing costs and time-lines significantly. Outsourcing software testing can cut costs significantly, improve quality and lower risks.

Vertical-specific legal and regulatory frameworks are becoming more stringent. In terms of technology absorption, the leading verticals have been banking, financial services and health-care. These verticals have witnessed the tightening of regulatory norms, which has meant significant investment in technology. HIPAA (for health care), BASEL II, SOX, anti-money laundering (on the banking and financial services) have been on top of the list. Even in the domestic market, the Reserve Bank of India has led the charge with implementation of an RTGS (Real Time Gross Settlement) system, which has set the pace for banks to increase significantly their technology deployment.

Technology sourcing strategies are more complex. Multiple vendors working on engagements are common because business requirements are more complex today. It is common to find banks with more than a hundred bought out applications, which are all integrated and drawing upon one another. With software getting more integrated within and outside the organization, it is increasing the dependence on the same for all business decisions.

Defects could result in both financial and non-financial (image, response time and so on) setbacks. In industries such as banking, where money is business and managing risks is critical, the effect of software defects can be disastrous. Independent testing is increasingly giving that additional assurance to clients.

Increasingly users of technology are non-technical and, therefore, their tolerance for poor quality is low. The quality of application going into production has become a key determinant of the evaluation of CIO's. The questions that need immediate answers are: If the new solution is improving the life of the business user? Is the solution making him more productive in delivering his role? Is it making the customer's life easier?

With these developments, testing is increasingly being seen as a specialized service. The presence of sound testing service providers in offshore locations, such as India, will help global corporate mitigate risks associated with off shoring. Today, testing is attracting better quality talent. Five years ago, testing was seen as a poor cousin to development. Today, both billing rates and compensation levels of testing professionals are comparable, if not better, than programming professional.

Lack of training academies that focus on testing is a serious issue today. Twenty years ago there were training institutes for computing languages. Knowledge of the SDLC and programming languages have now found their way into school and college curricula. Testing is yet to get the attention of academic institutions. Given the promising rate of growth in testing that took over the last few years, the next five years will see more specialized programmes around software testing which would further strengthen the position of software testing as a sound career option.

CHENNAI AS A TESTING HUB

The current market size and the estimated growth (at an compound annual growth rate (CAGR) of 92 percent) have created a lot of excitement in the industry. In India, Chennai is fast growing as a major software-testing destination. "The city is undoubtedly emerging as the software testing hub of India," says Vanaja Arvind, Executive Director, Think soft Global Services, one of India's leading standalone and domain-focused independent software testing companies.

While the early entrants - Think soft, Maveric, Amity Soft, RTG, etc. - leading the band, relatively new players are also doing a booming business. Trimentus Technologies, established in 2002, has done a million dollar business the previous year and looks for a two-and-a-half times growth this year. Standardization Testing Quality Certification (STQC), a significant player in testing, has set up its testing facility in Chennai recently.

Also, large IT service providers are adding the testing services in their service offerings. For instance, Cognizant Technology Services is strengthening its testing activity. "Testing is one of the fastest growing service offerings for Cognizant," says G Sumitra, Director, Testing Practice. The company is seeing significant traction in its testing practice. In 2004, the number of career testers of Cognizant grew from 200 to over 1000 - the number is expected to double in 2005.

The world leaders in testing like Mercury and RelQ have set up their offshore base in India. Mercury India, the Indian arm of Mercury Interactive Corporation, which is into business technology optimization (BTO), says that its Indian centre would cater to the application delivery and application management market, which is expected to emerge as one of the most strategic areas of IT investments. To be successful globally and garner a large share in the outsourcing market, the players need to focus on strengthening their capabilities in automated testing, which is a must to support a number of mission critical processes.

BANGALORE, INDIA

India is poised to become a leader in the software testing market with an increasing number of companies outsourcing their software testing work to India. Industry analyst Gartner Group finds that testing could make up to 25-50 per cent of software budgets. Of this, independent testing is growing at 50-65 per cent while the part of work done offshore is growing at 35-40 per cent. Another study reveals that the software testing market in India is pegged to touch the USD 2 billion mark in 2007 itself.

India, already being a huge destination for software development, makes it logical for independent software vendors (ISVs) to offshore testing to India resulting in significant cost savings and reducing time-to-market as it reduces the overall product lifecycle. Regulatory bodies in the US & UK are getting more stringent with compliance norms putting an upward thrust to the global software testing market. ISVs are also keen to get their products tested by a third party which has played little or no role in the software development process, also serving as beta users for the products in the process. There is an increasing pressure from the user community also demanding that vendors take up the onus for vulnerabilities in their products, thus the demand for testing of software products is ought to rise further. Organizations are also hiring testing service providers to validate the products (ERP, CRM, etc) which they are implementing.

Testing services in India is increasingly being seen as a specialized offering considering the fact that IT companies have started positioning themselves as independent software testing (IST) services providers, providing software testing as a specialized service. Testing service market in India is looking upwards with the rapid expansion of these IST firms in the country. Testing can now be compared with other IT services in terms of revenue for companies and also in terms of remuneration it is offering. In addition to the pure IST players which are rapidly coming up, majority of the top companies are

setting up IST centers to cater to this booming market. Software testing in India is entering into the next level of maturity. IT professionals in the country are experiencing a change in perception in terms of their approach to software testing and are now considering it as a serious career option.

Testing requires extensive domain knowledge and is being considered as an area of specialization in areas like real-time testing, web-applications, web services, compliance, performance, inter-operability, embedded systems, etc to name a few. It is estimated that industry requires around 25,000 to 30,000 testing professionals within a year. To bridge this gap an increasing number of non-IT graduates are being inducted by pure ISTs and even large IT companies.

Software testing is climbing up the priority list of global product companies which is contributing to the increase in off shoring testing services to India. Adding to this is the growth in the domestic software testing market with growth in sectors like BFSI, insurance, government, manufacturing, etc. India Inc. needs to continue its strengthening of capabilities in automated testing and build a pool of skilled professionals with domain and tool expertise to leverage the huge potential in this industry.

4. DATA ANALYSIS AND INTERPRETATION

CHAPTER 4

DATA ANALYSIS AND INTERPRETATION

4.1 TEST CASE REPORT ANALYSIS

TABLE No. 4.1.1

PERCENTAGE OF SYSTEM AND INTEGRATION TESTS IN JOB ORDER SYSTEM MODULE

TEST TYPE	NO OF TESTS	PERCENTAGE
SYSTEM TEST	91	94
INTEGRATION TEST	6	6
TOTAL	97	100

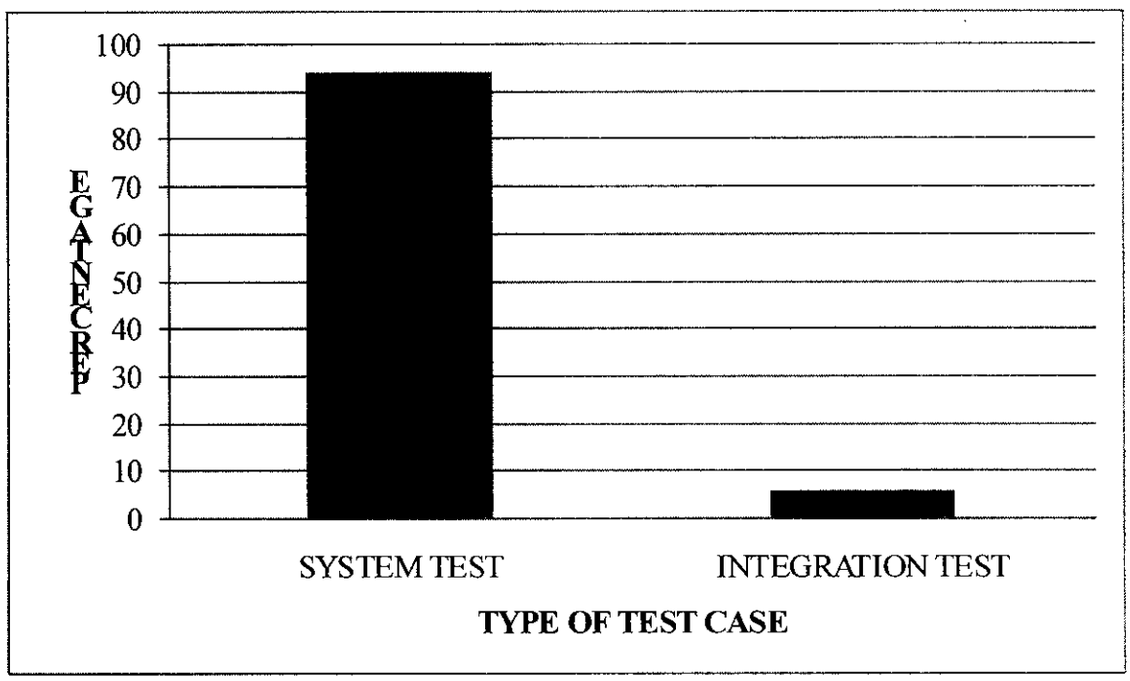


CHART No. 4.1.1 PERCENTAGE OF SYSTEM AND INTEGRATION TESTS IN JOB ORDER SYSTEM MODULE

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 94 % of system tests cases and 6% of integration test cases in the job order module

TABLE No. 4.1.2

PERCENTAGE OF SYSTEM AND INTEGRATION TESTS IN PRODUCTION PROCESS MODULE

TEST TYPE	NO OF TESTS	PERCENTAGE
SYSTEM TEST	131	92
INTEGRATION TEST	11	8
TOTAL	142	100

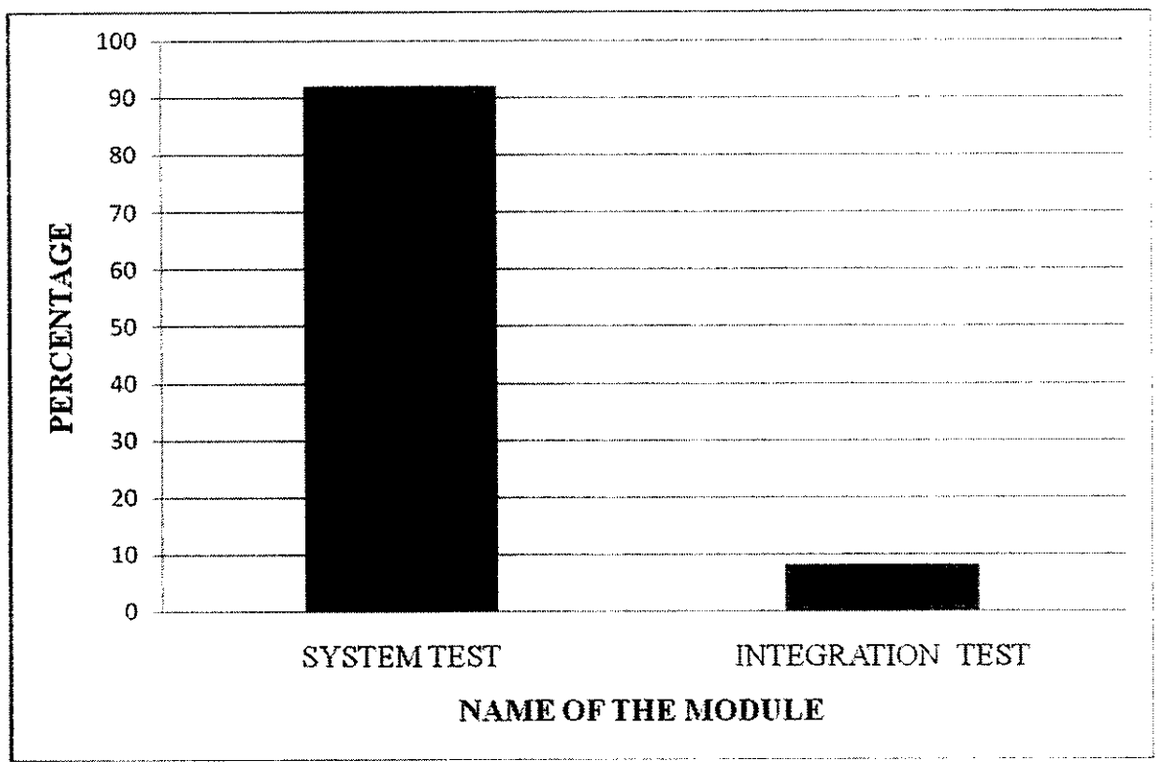


CHART No. 4.1.2 PERCENTAGE OF SYSTEM AND INTEGRATION TESTS IN PRODUCTION PROCESS MODULE

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 92% of system tests cases and 8% of integration test cases in the production process module.

TABLE No. 4.1.3

PERCENTAGE OF SYSTEM TESTS IN BOTH THE MODULES

MODULES	NO OF SYSTEM TESTS	PERCENTAGE
JOB ORDER SYSTEM	91	41
PRODUCTION PROCESS	131	59
TOTAL	222	100

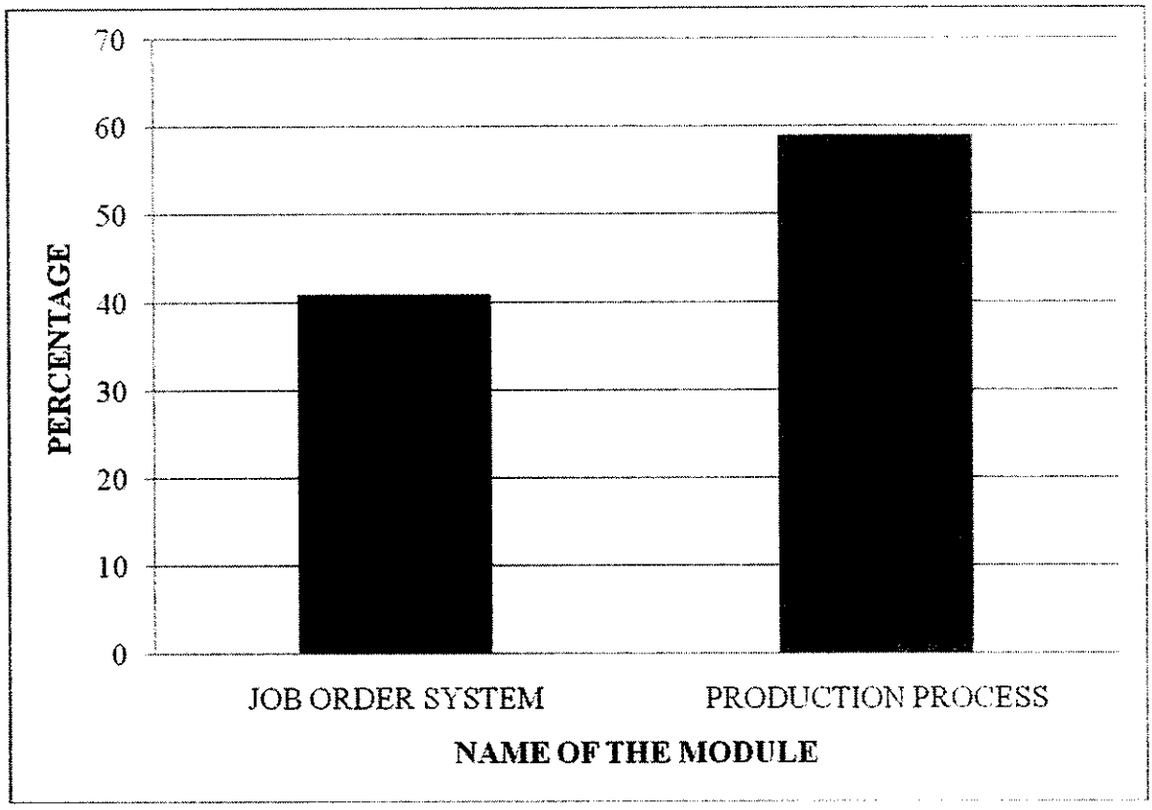


CHART No. 4.1.3 PERCENTAGE OF SYSTEM TESTS IN BOTH THE MODULES

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 41% of system tests cases in Job Order module and 59% of system test cases in the production process module.

TABLE No. 4.1.4

PERCENTAGE OF INTEGRATION TESTS IN BOTH THE MODULES

MODULES	NO OF INTEGRATION TESTS	PERCENTAGE
JOB ORDER SYSTEM	6	35
PRODUCTION PROCESS	11	65
TOTAL	17	100

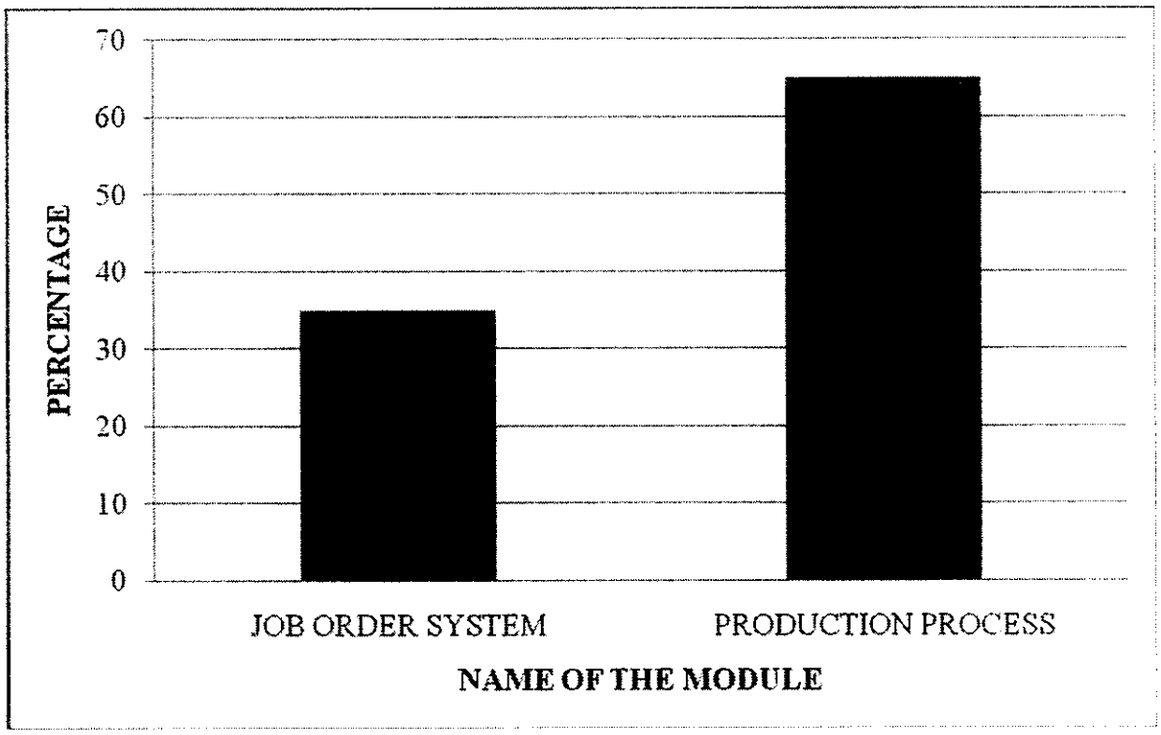


CHART No. 4.1.4 PERCENTAGE OF INTEGRATION TESTS IN BOTH THE MODULES

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 35% of integration test cases in Job Order module and 65% of integration test cases in the production process module.

TABLE No. 4.1.5

PERCENTAGE OF POSITIVE &NEGATIVE TEST CASES IN JOB ORDER SYSTEM MODULE

TEST CASES TYPE	NO OF TESTS	PERCENTAGE
POSITIVE TEST CASES	69	71
NEGATIVE TEST CASES	28	29
TOTAL	97	100

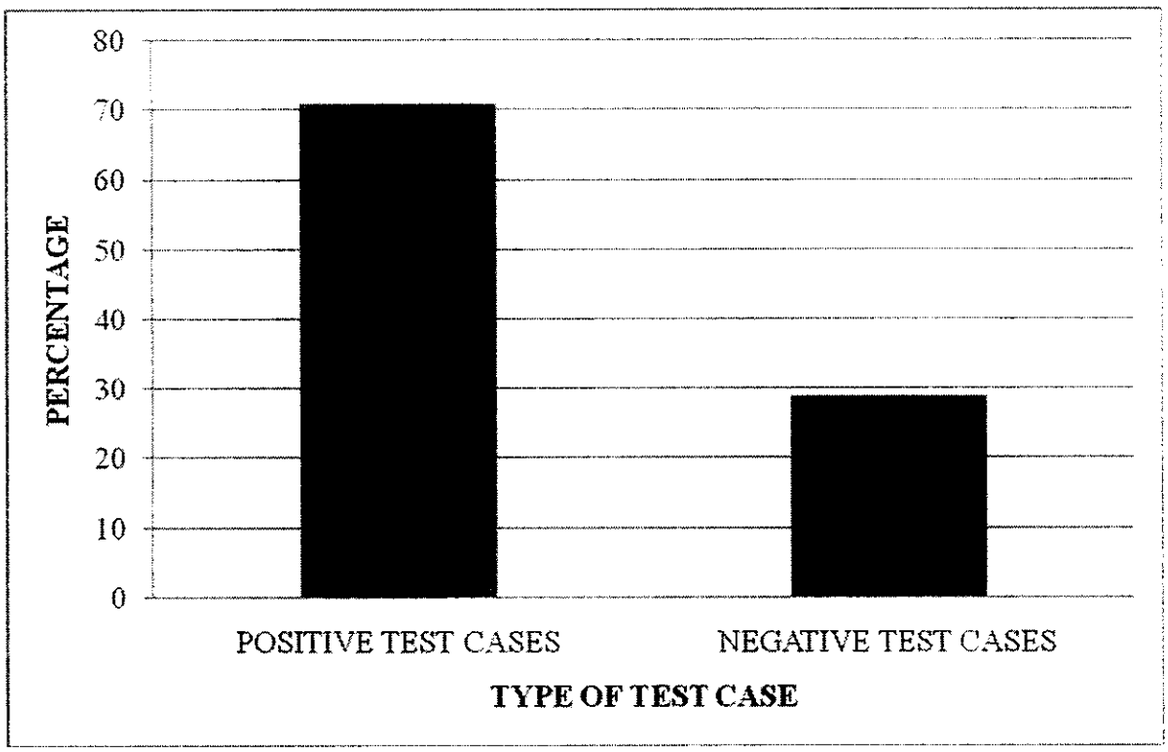


CHART No. 4.1.5 PERCENTAGE OF POSITIVE &NEGATIVE TEST CASES IN JOB ORDER SYSTEM MODULE

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 71% of positive test cases and 29 % of negative test cases in the job order system module

TABLE No. 4.1.6

PERCENTAGE OF POSTIVE & NEGATIVE TEST CASES IN PRODUCTION PROCESS MODULE

TEST CASE TYPE	NO OF TEST CASES	PERCENTAGE
POSITIVE TEST CASES	87	58
NEGATIVE TEST CASES	62	42
TOTAL	149	100

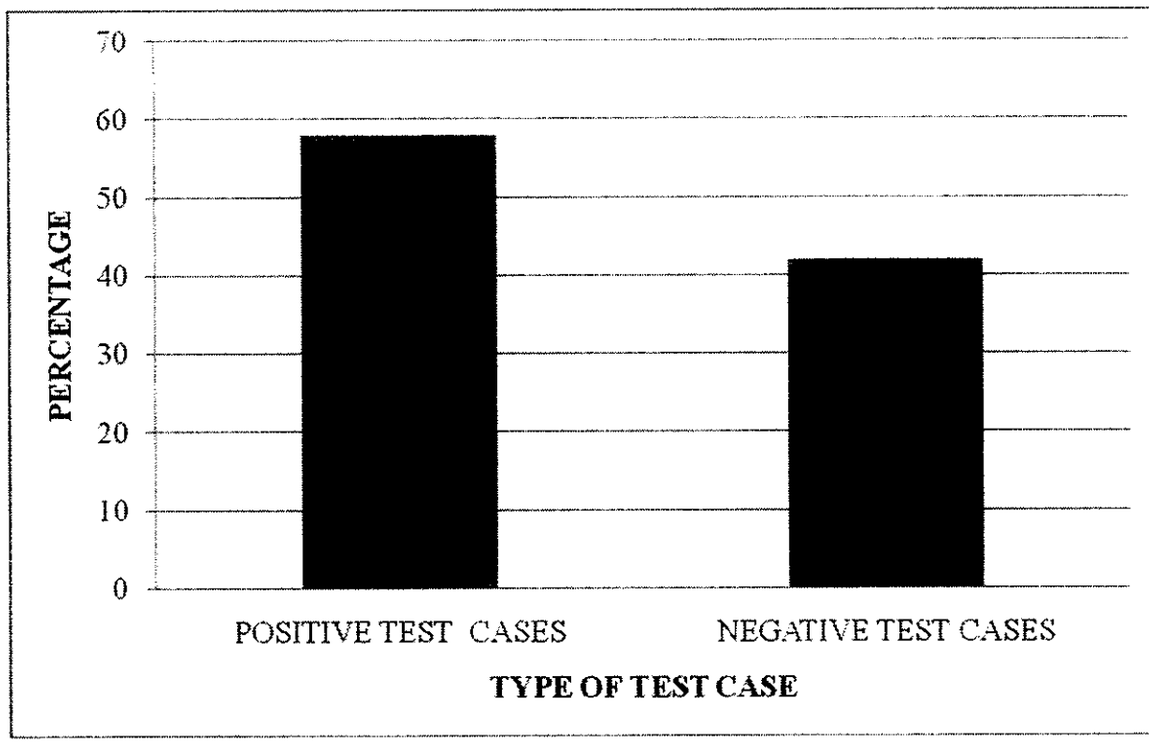


CHART No. 4.1.6 PERCENTAGE OF POSTIVE & NEGATIVE TEST CASES IN PRODUCTION PROCESS MODULE

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 58% of positive test cases and 42 % of negative test cases in the production process system module

TABLE No. 4.1.7

PERCENTAGE OF POSITIVE TEST CASES IN BOTH THE MODULES

MODULES	NO OF POSITIVE TEST CASES	PERCENTAGE
JOB ORDER SYSTEM	69	44
PRODUCTION PROCESS	87	56
TOTAL	156	100

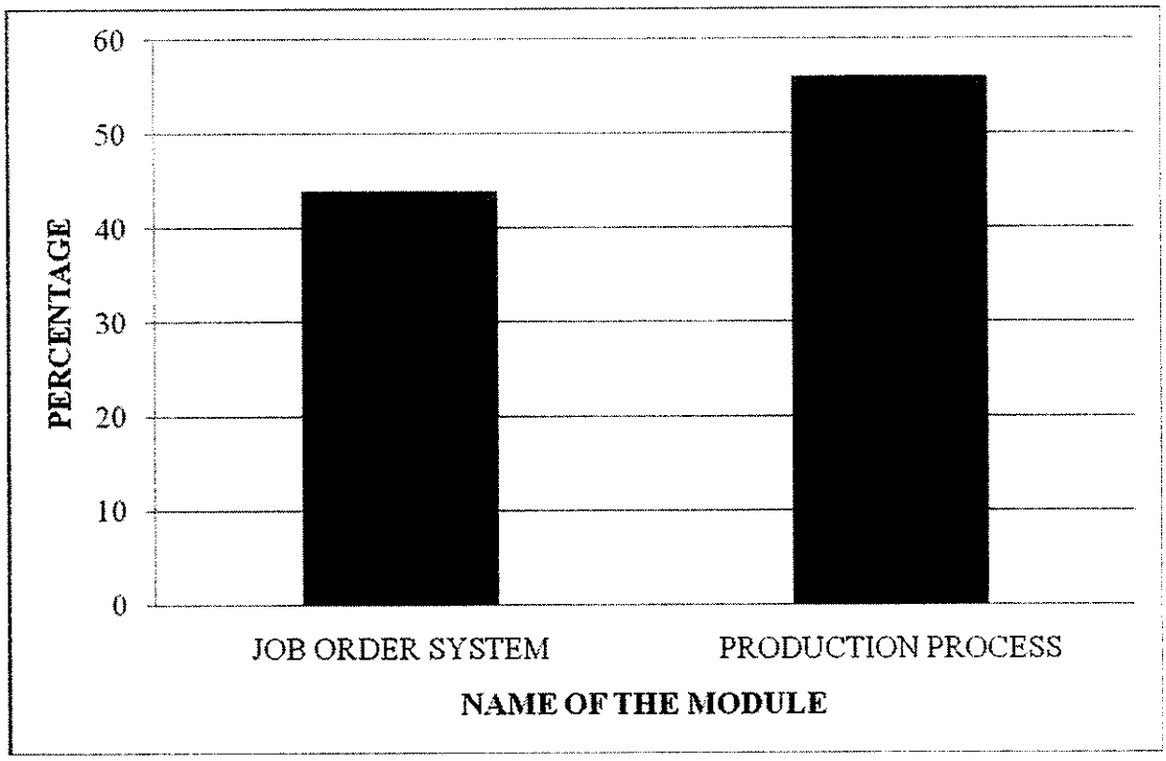


CHART No. 4.1.7 PERCENTAGE OF POSITIVE TEST CASES IN BOTH THE MODULES

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 44% of positive test cases in job order module and 56 % of positive test cases in the production process system module

TABLE No. 4.1.8

PERCENTAGE OF NEGATIVE TEST CASES IN BOTH THE MODULES

MODULES	NO OF NEGATIVE TEST CASES	PERCENTAGE
JOB ORDER SYSTEM	28	24
PRODUCTION PROCESS	87	76
TOTAL	115	100

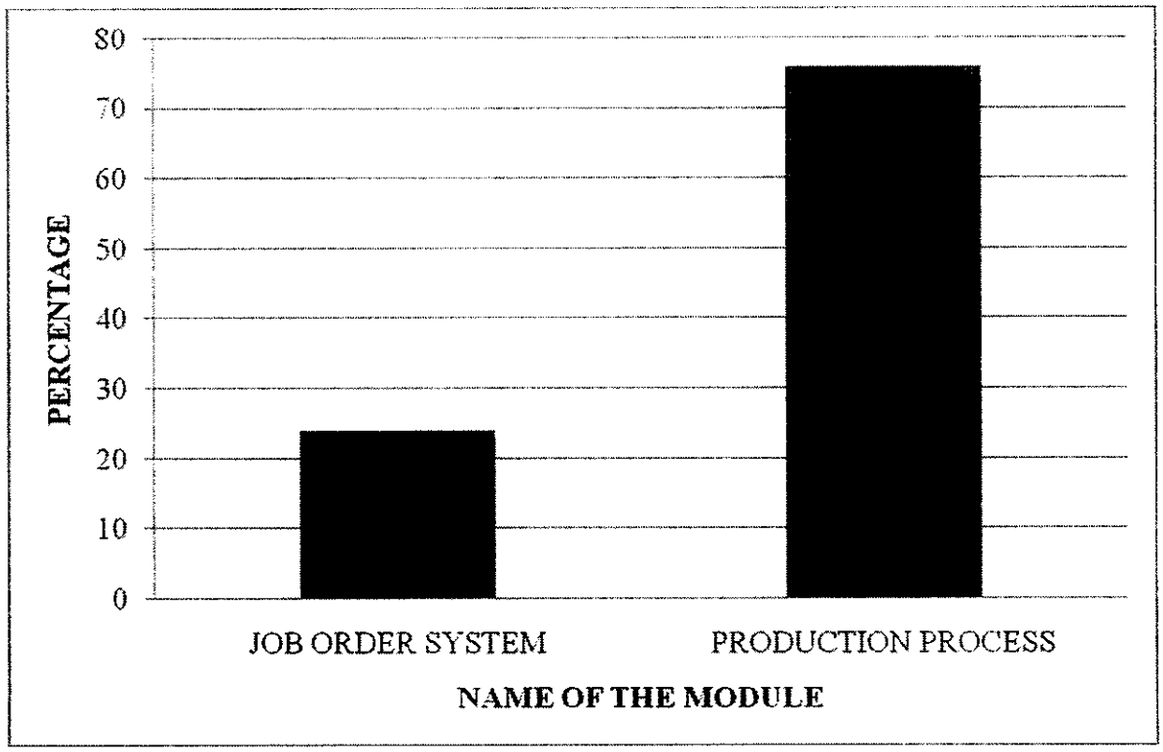


CHART No. 4.1.8 PERCENTAGE OF NEGATIVE TEST CASES IN BOTH THE MODULES

INTERPRETATION:

From the above figure it can be seen that the test data store consists of 24% of negative test cases in job order module and 76 % of negative test cases in the production process system module

TABLE No. 4.1.9

PERCENTAGE OF PASSED AND FAILED TEST CASES IN JOB ORDER SYSTEM MODULE

TEST CASE TYPE	NO OF TESTCASES	PERCENTAGE
PASSED TEST CASES	92	95
FAILED TEST CASES	5	5
TOTAL	97	100

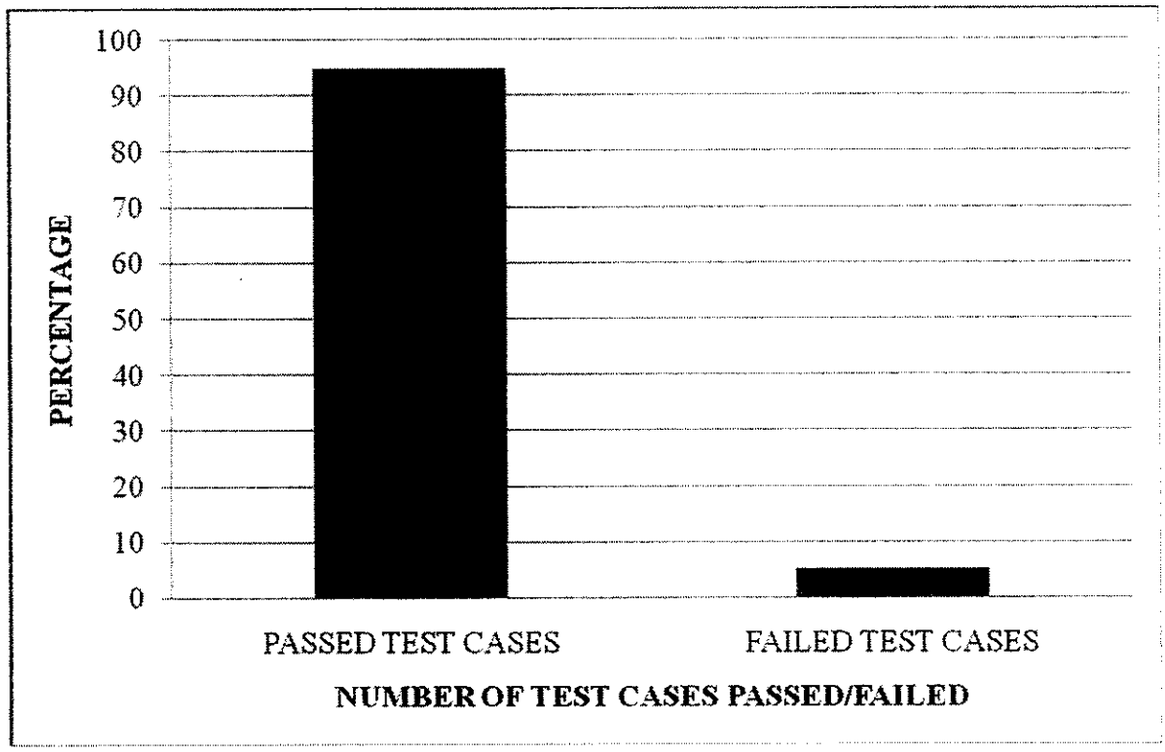


CHART No. 4.1.9 PERCENTAGE OF PASSED AND FAILED TEST CASES IN JOB ORDER SYSTEM MODULE

INTERPRETATION:

From the above figure it can be seen that the job order system module consists of 95% of passed test cases and only 5% of failed test cases. Therefore this signifies that the number of defects in the job order system module is very low

TABLE No. 4.1.10

PERCENTAGE OF PASSED & FAILED TEST CASES IN PRODUCTION PROCESS MODULE

TEST CASE TYPE	NO OF TESTCASES	PERCENTAGE
PASSED TESTCASES	136	91
FAILED TEST CASES	13	9
TOTAL	149	100

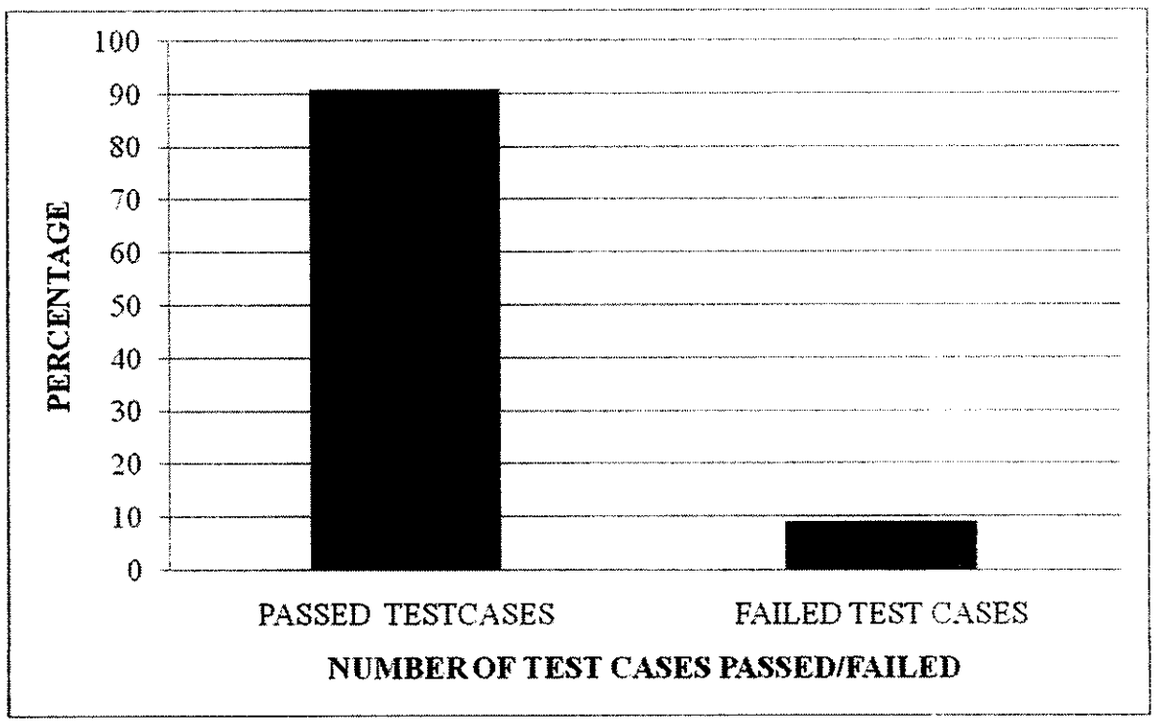


CHART No. 4.1.10 PERCENTAGE OF PASSED & FAILED TEST CASES IN PRODUCTION PROCESS MODULE

INTERPRETATION:

From the above figure it can be seen that the production process module consists of 91% of passed test cases and only 9% of failed test cases. Therefore this signifies that the number of defects in the production process module is very low but it is relatively high when compared to the defects in the job order system module.

TABLE No. 4.1.11

PERCENTAGE OF PASSED TEST CASES IN BOTH THE MODULES

MODULES	NO OF PASSED TEST CASES	PERCENTAGE
JOB ORDER SYSTEM	92	40
PRODUCTION PROCESS	136	60
TOTAL	228	100

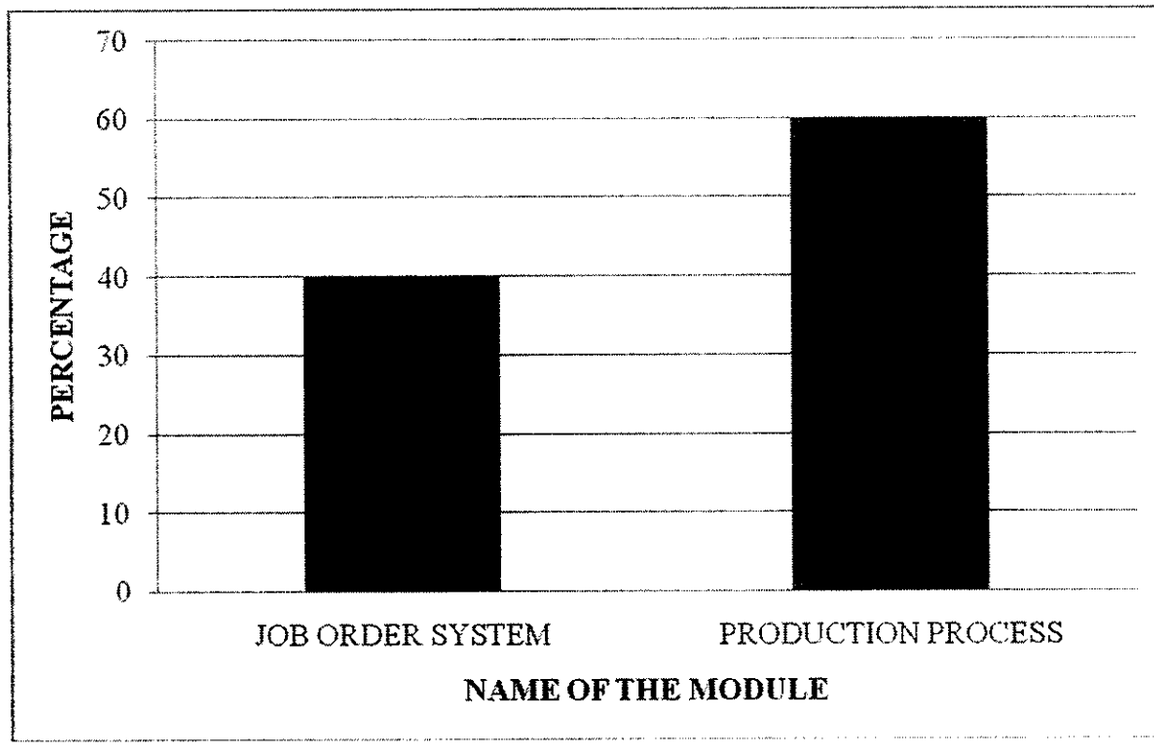


CHART No. 4.1.11PERCENTAGE OF PASSED TEST CASES IN BOTH THE MODULES

INTERPRETATION:

From the above figure it can be seen that the job order system module consists of 40% of passed test cases and 60% of failed test cases. This denotes that the number of defects in the job order system module is lesser than the defects in the production process module

TABLE No. 4.1.12

PERCENTAGE OF FAILED TEST CASES IN BOTH THE MODULES

MODULES	NO OF FAILED TEST CASES	PERCENTAGE
JOB ORDER SYSTEM	5	28
PRODUCTION PROCESS	13	72
TOTAL	18	100

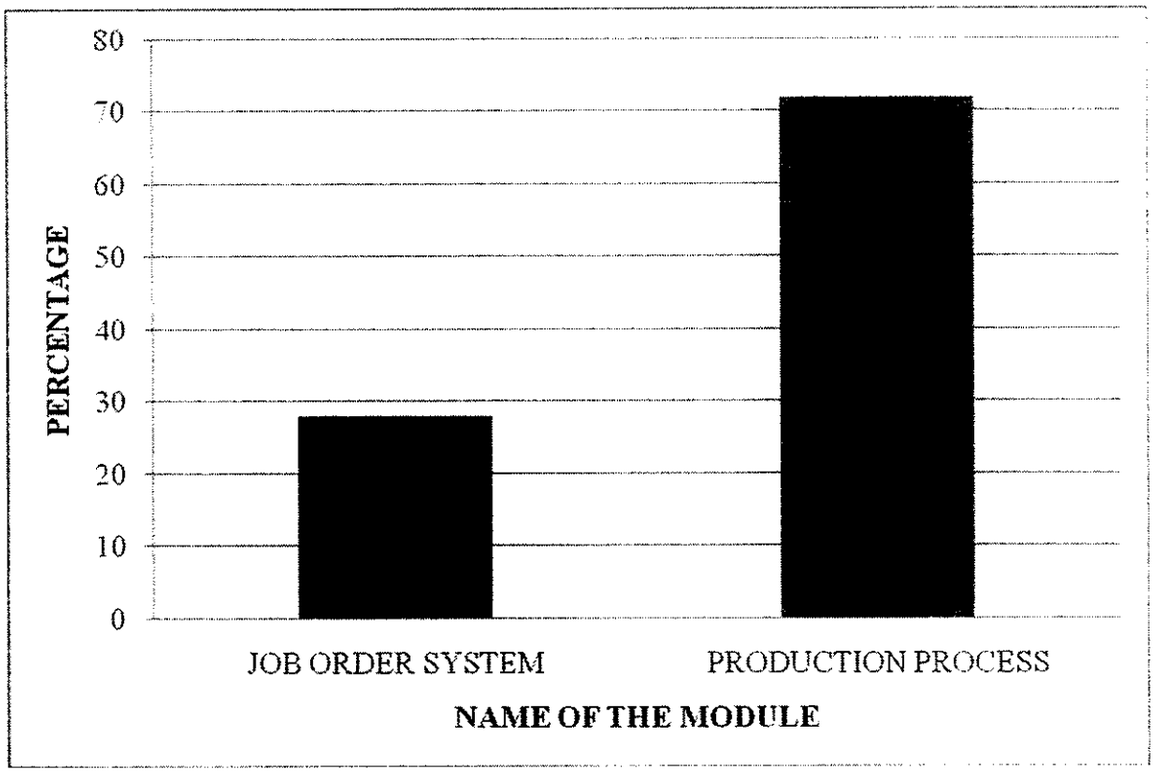


CHART No. 4.1.12 PERCENTAGE OF FAILED TEST CASES IN BOTH THE MODULES

INTERPRETATION:

From the above figure it can be seen that the job order system module consists of 28% of passed test cases and 72% of failed test cases. This denotes that the production process module contributes to more number of defects in the production process and control system project

4.2 PEER REVIEW ANALYSIS

TABLE No. 4.2.1

OPINION OF THE RESPONDENTS REGARDING THE SPECIFICATION OF VERSION OF THE SCREEN SHOT

OPINION OF RESPONDENTS	No. OF RESPONDENTS	PERCENTAGE
YES	9	60
NO	6	40
TOTAL	15	100

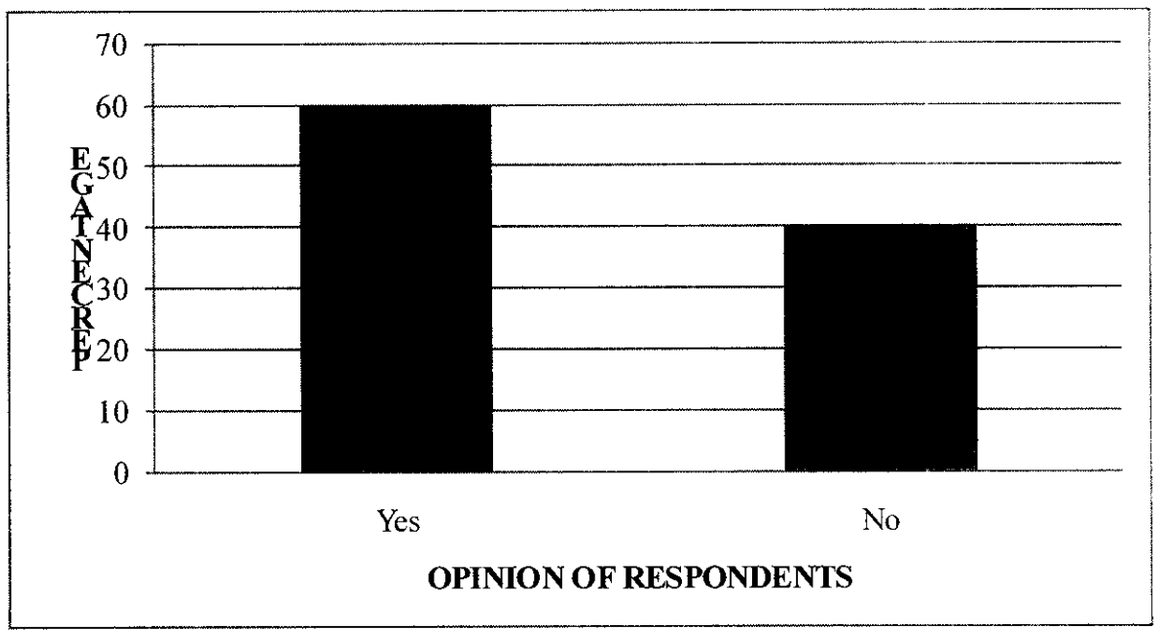


CHART No: 4.1.13 OPINION OF THE RESPONDENTS REGARDING THE SPECIFICATION OF VERSION OF THE SCREEN SHOT

INTERPRETATION:

From the above table it is seen that 60% of respondents are satisfied with the specification of the version of screen shot but the rest 40% are dissatisfied with it.

TABLE No. 4.2.2

OPINION OF THE RESPONDENTS REGARDING THE ACCURACY OF THE TEST CASES

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	13	87
NO	2	13
TOTAL	15	100

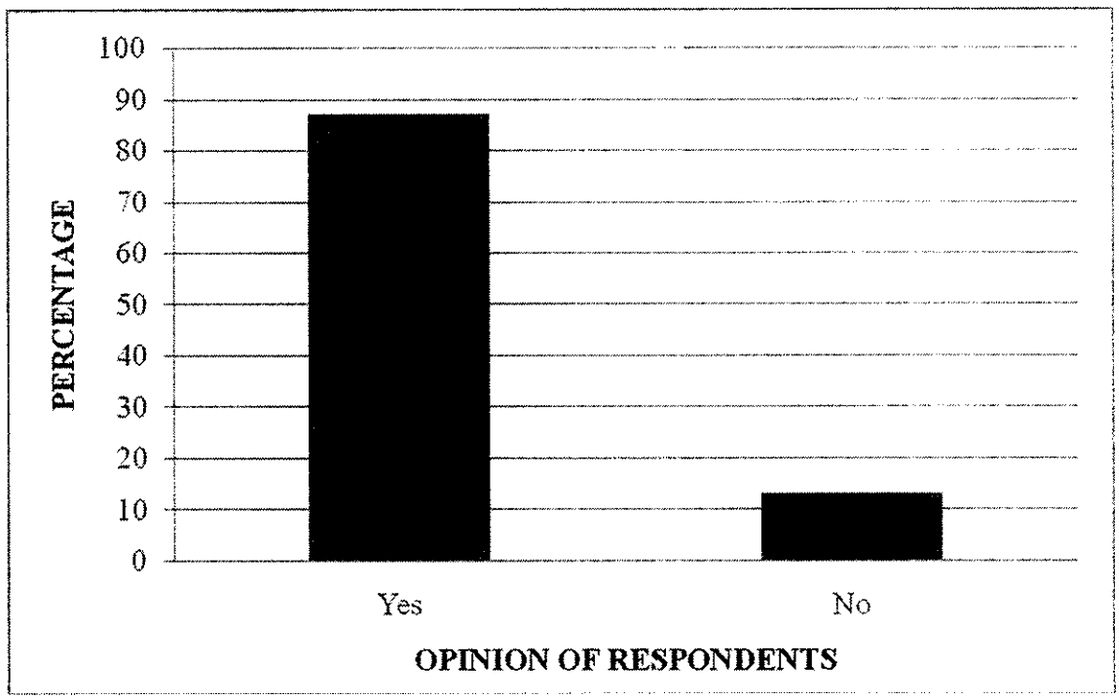


CHART No. 4.2.2 OPINION OF THE RESPONDENTS REGARDING THE ACCURACY OF THE TEST CASES

INTERPRETATION:

From the above table it is seen that 87% of respondents are satisfied with the accuracy of test case description but the rest 13% are dissatisfied with it.

TABLE No. 4.2.3

OPINION OF THE RESPONDENT REGARDING THE CLEAR MENTIONING OF THE TEST CASE FUNCTIONALITY

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	14	93
NO	1	7
TOTAL	15	100

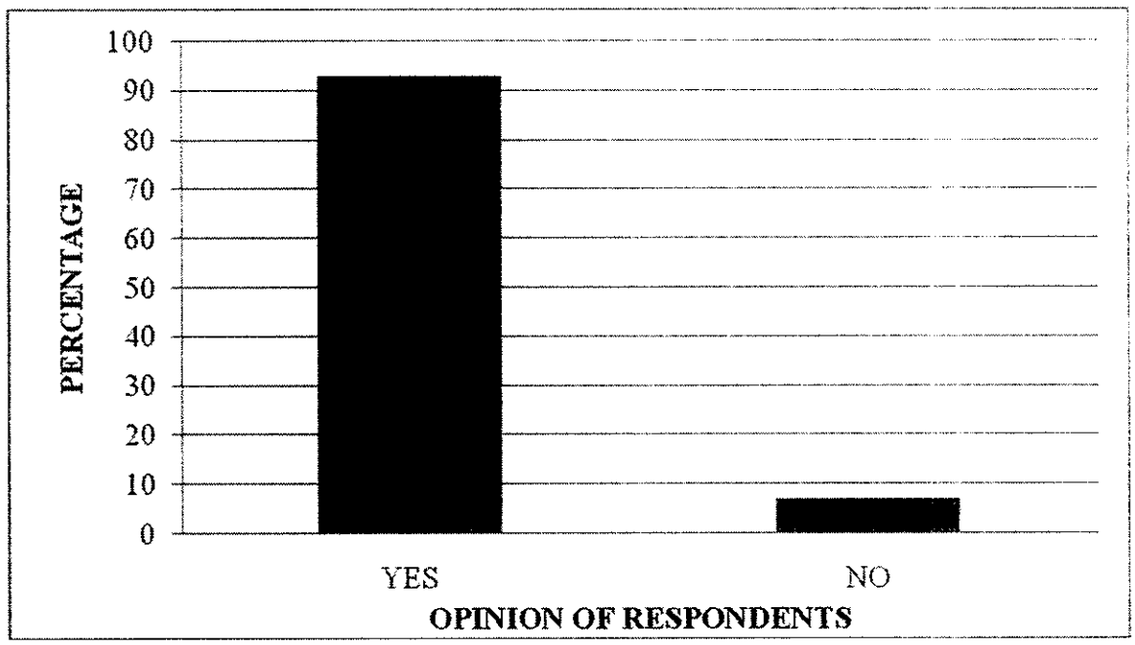


CHART No. 4.2.3 OPINION OF THE RESPONDENT REGARDING THE CLEAR MENTIONING OF THE TEST CASE FUNCTIONALITY

INTERPRETATION:

From the above table it is seen that 93% of respondents are satisfied with the clarity in the mentioning of the test case functionality but the rest 7% are dissatisfied with it.

TABLE No. 4.2.4

OPINION OF THE RESPONDENT REGARDING THE CLEAR MENTIONING OF THE TEST CASE ACTIONS

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	14	93
NO	1	7
TOTAL	15	100

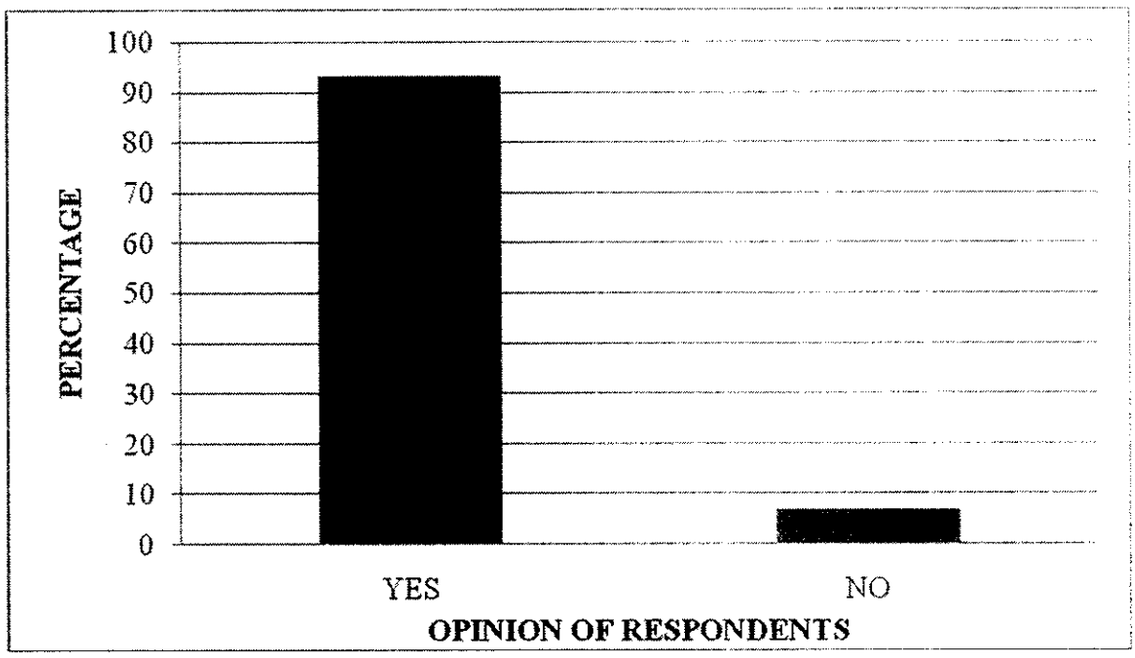


CHART No: 4.2.4 OPINION OF THE RESPONDENT REGARDING THE CLEAR MENTIONING OF THE TEST CASE ACTIONS

INTERPRETATION:

From the above table it is seen that 93% of respondents are satisfied with the clarity in the mentioning of the test case actions but the rest 7% are dissatisfied with it.

TABLE No. 4.2.5

OPINION OF THE RESPONDENT REGARDING THE DESCRIPTION OF THE EXPECTED RESULTS

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	13	87
NO	2	13
TOTAL	15	100

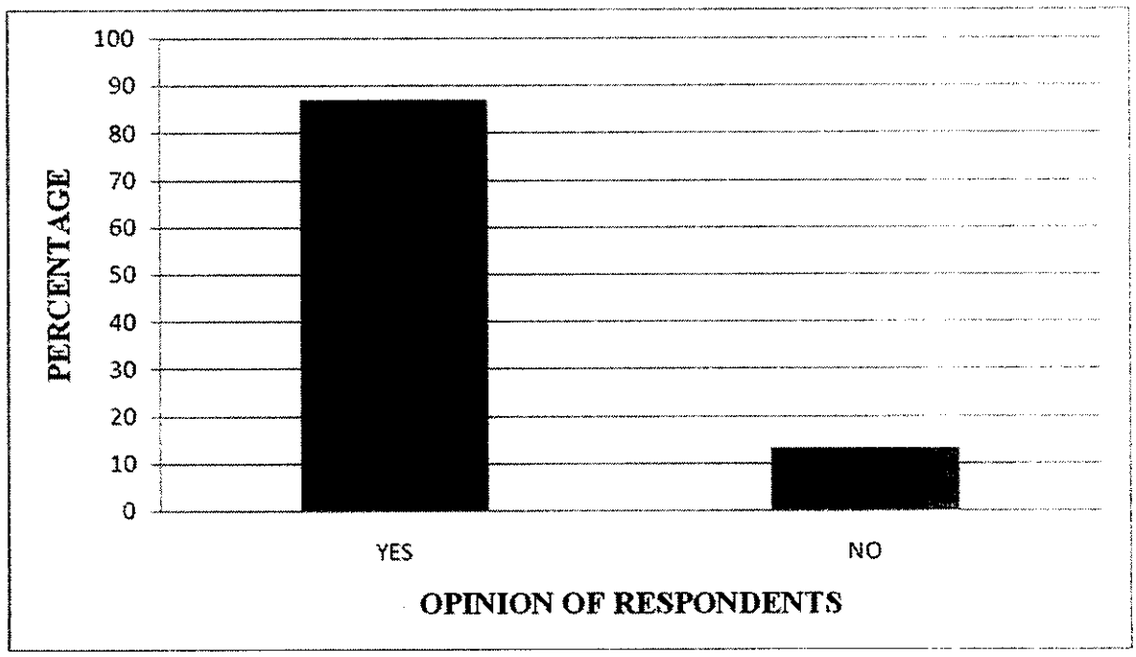


CHART No. 4.2.5 OPINION OF THE RESPONDENT REGARDING THE DESCRIPTION OF THE EXPECTED RESULTS

INTERPRETATION:

From the above table it is seen that 87% of respondents have shown a positive response with the clarity in the mentioning of the test case functionality but the rest 13% have shown a negative response

TABLE No. 4.2.6

OPINION OF THE RESPONDENT REGARDING THE CLARITY AND UNAMBIGUITY OF THE SUCCESS/FAILURE CRITERIA

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	14	93
NO	1	7
TOTAL	15	100

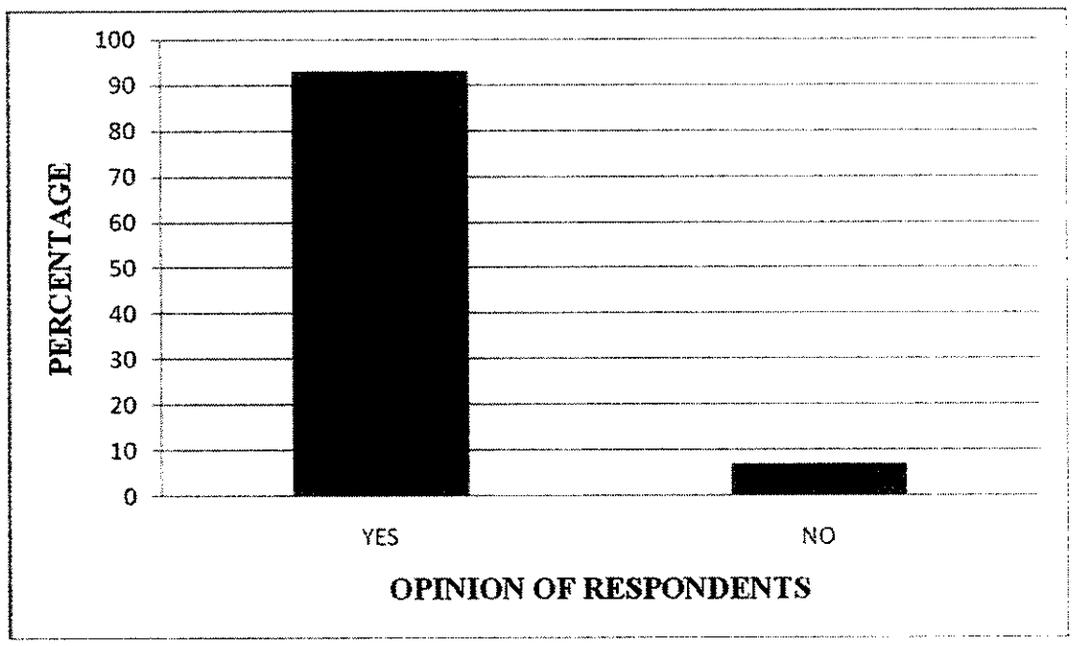


CHART No. 4.2.6 OPINION OF THE RESPONDENT REGARDING THE CLARITY AND UNAMBIGUITY OF THE SUCCESS/FAILURE CRITERIA

INTERPRETATION:

From the above table it is seen that 93% of respondents have shown a positive response with the clarity and unambiguity of the success/failure criteria but the rest 7% have shown a negative response towards it

TABLE No. 4.2.7

OPINION OF THE RESPONDENT REGARDING THE DESCRIPTION OF THE EXPECTED RESPONSE OF INDIVIDUAL TEST CASE STEPS

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	12	80
NO	3	20
TOTAL	15	100

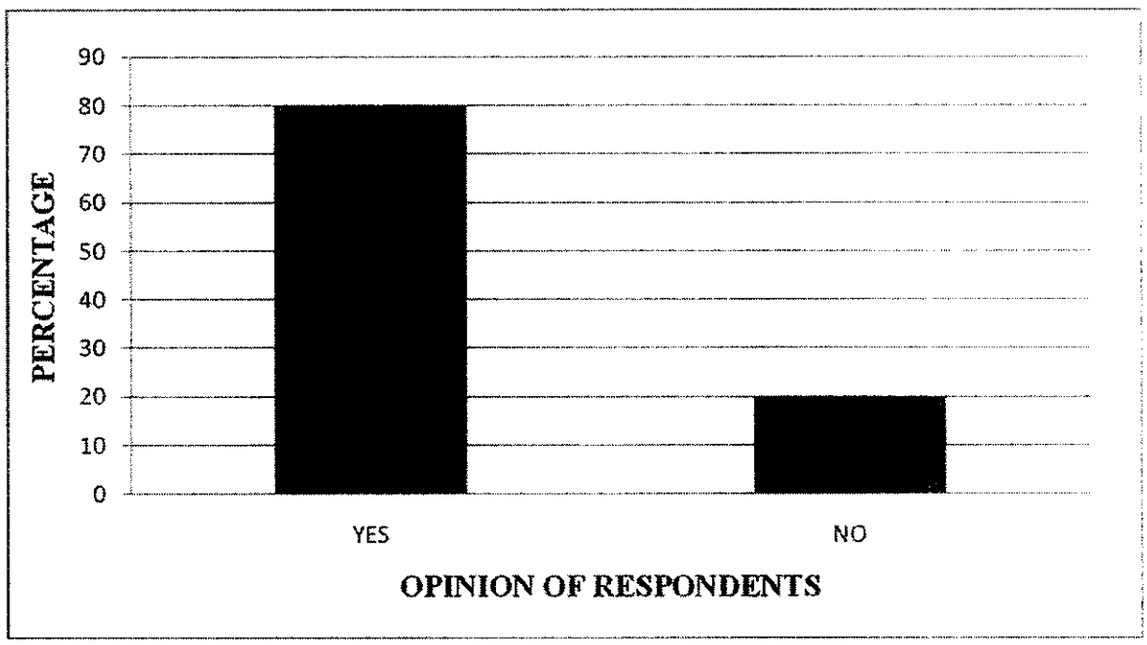


CHART No. 4.2.7 OPINION OF THE RESPONDENT REGARDING THE DESCRIPTION OF THE EXPECTED RESPONSE OF INDIVIDUAL TEST CASE STEPS

INTERPRETATION:

From the above table it is seen that 80% of respondents have shown a positive response with description of the expected response of individual test case steps but the rest 20% have shown a negative response towards it

TABLE No.4.2.8

OPINION OF THE RESPONDENT REGARDING THE PREPARATION OF TEST CASES FOR ALL SCENARIOS AND LOGICAL LOOPS

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	12	80
NO	3	20
TOTAL	15	100

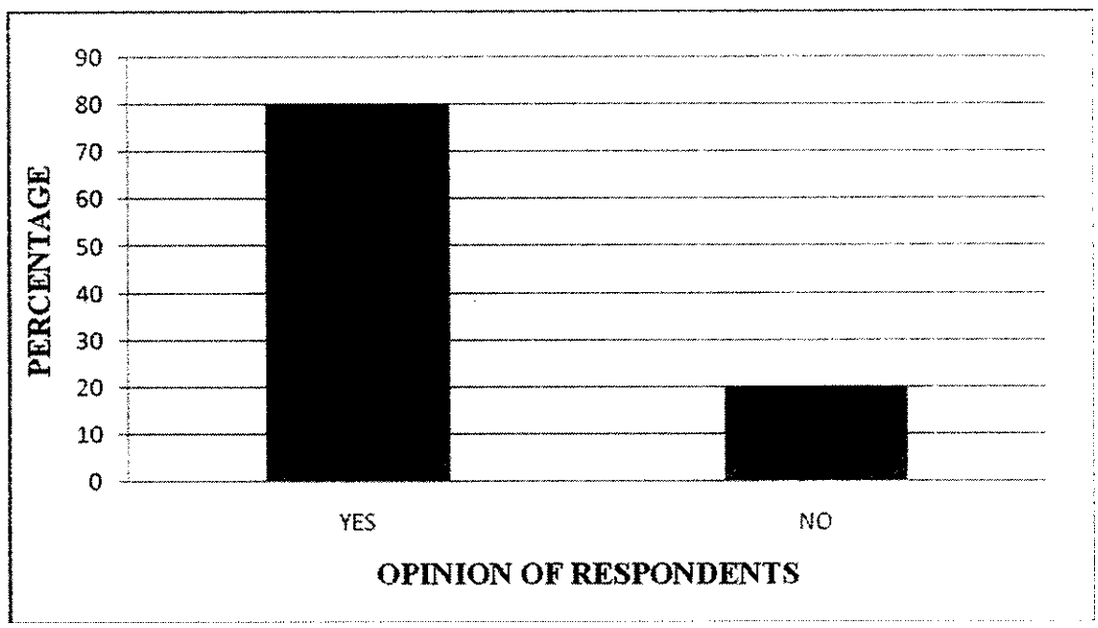


CHART No.4.2.8 OPINION OF THE RESPONDENT REGARDING THE PREPARATION OF TEST CASES FOR ALL SCENARIOS AND LOGICAL LOOPS

INTERPRETATION:

From the above table it is seen that 80% of respondents have shown a positive response with the preparation of test cases for all scenarios and logical loops but the rest 20% have shown a negative response towards it

TABLE No.4.2.9

OPINION OF THE RESPONDENT REGARDING THE IDENTIFICATION OF TEST CASE DEPENDENCIES

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	11	73
NO	4	27
TOTAL	15	100

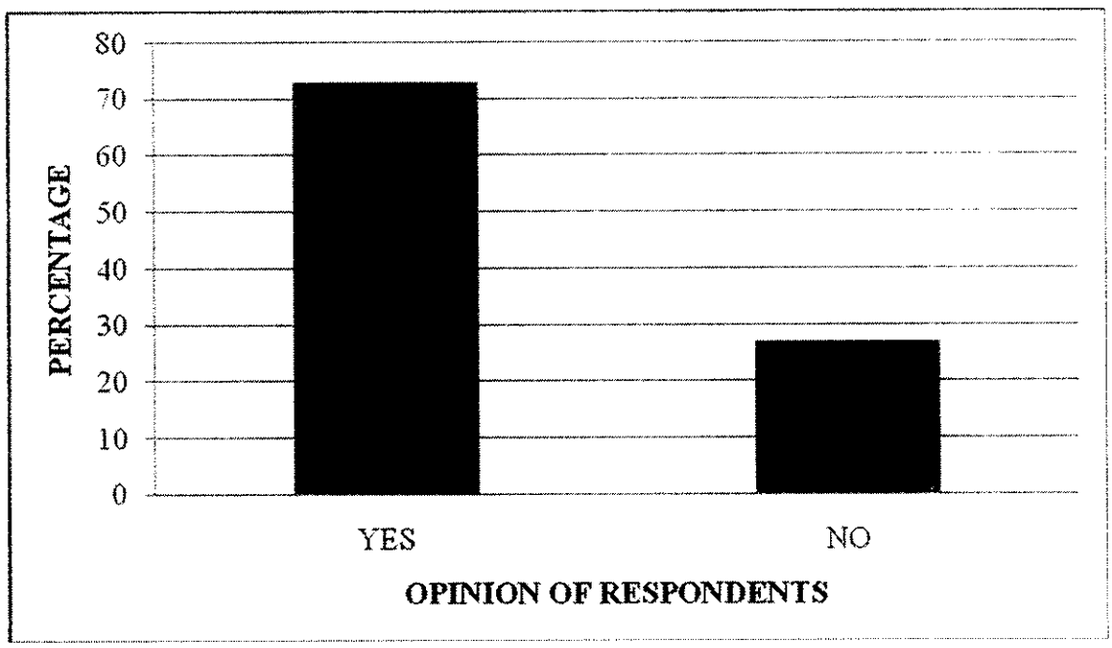


CHART No. 4.2.9 OPINION OF THE RESPONDENT REGARDING THE IDENTIFICATION OF TEST CASE DEPENDENCIES

INTERPRETATION:

From the above table it is seen that 73% of respondents have shown a positive response with identification of test case dependencies but the rest 27% have shown a negative response towards it

TABLE No. 4.2.10

OPINION OF THE RESPONDENT AS TO WHETHER THE EXPECTED RESULTS OF THE TEST CASES AGREE WITH THE EXPECTED PROGRAM BEHAVIOUR

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	14	93
NO	1	7
TOTAL	15	100

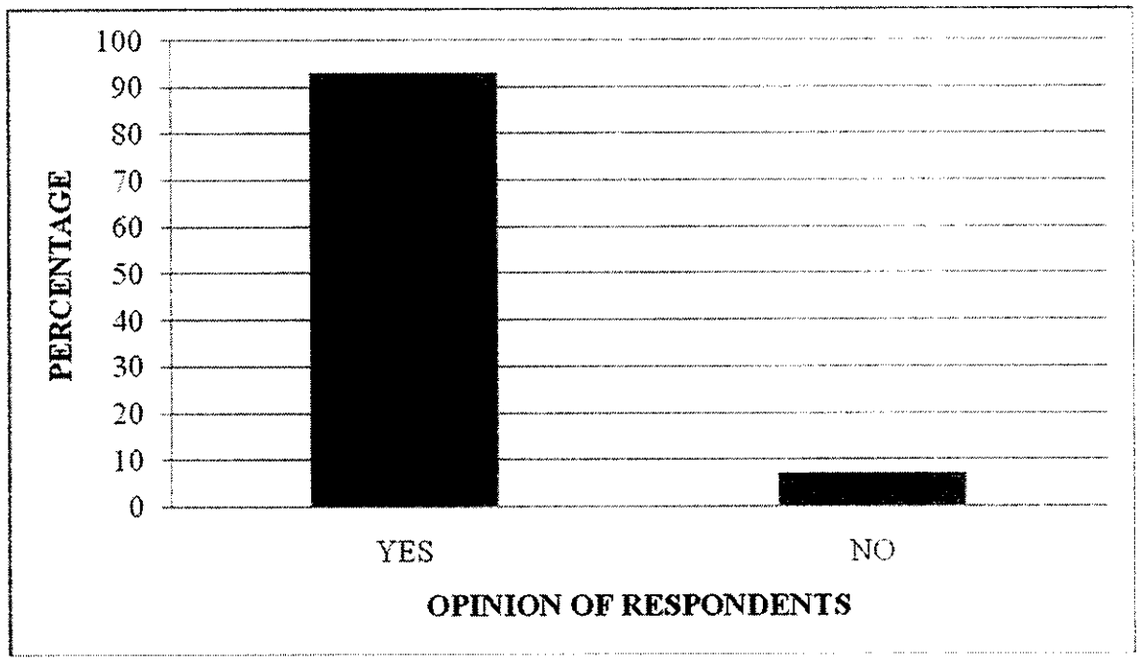


CHART No.4.2.10 OPINION OF THE RESPONDENT AS TO WHETHER THE EXPECTED RESULTS OF THE TEST CASES AGREE WITH THE EXPECTED PROGRAM BEHAVIOUR

INTERPRETATION:

From the above table it is seen that 93% of respondents have shown their satisfaction that the expected results of test cases agree with the expected program behavior but the rest 7% have shown discontent towards it

TABLE No. 4.2.11

OPINION OF THE RESPONDENT REGARDING THE CLASSIFICATION OF TEST CASES

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	9	60
NO	6	40
TOTAL	15	100

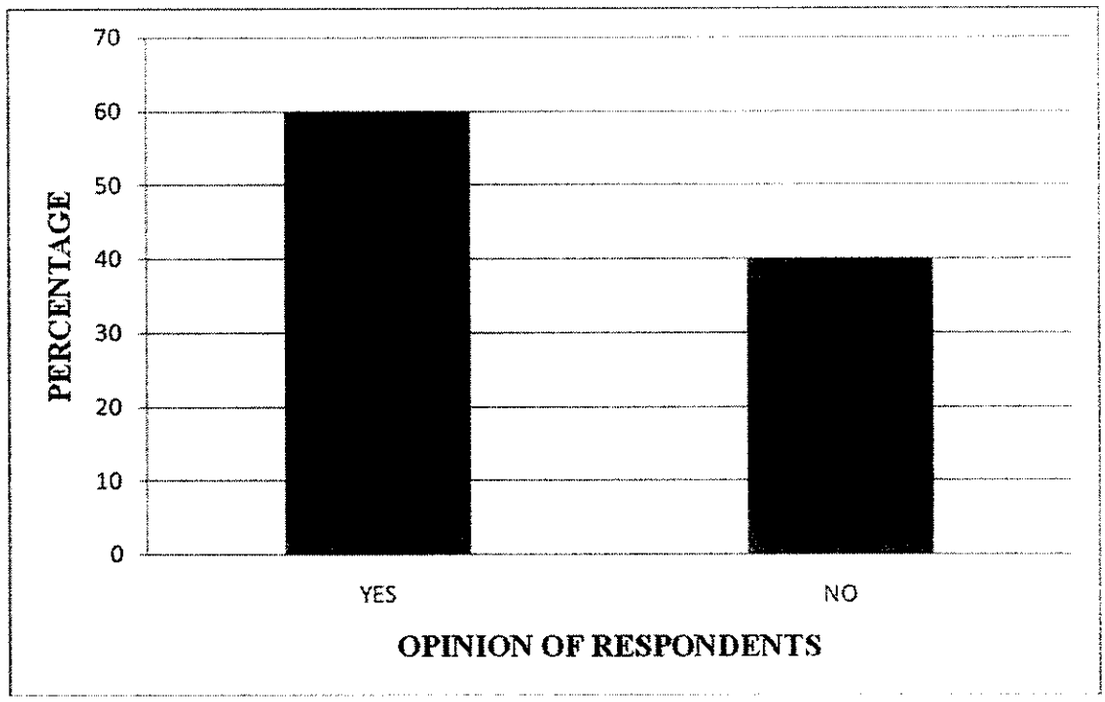


CHART No. 4.2.11 OPINION OF THE RESPONDENT REGARDING THE CLASSIFICATION OF TEST CASES

INTERPRETATION:

From the above table it is seen that 60% of respondents have shown a positive response with the classification of test cases but the rest 40% have shown a negative response towards it

TABLE No. 4.2.12

OPINION OF THE RESPONDENT REGARDING THE DESCRIPTION OF THE UNIT CASES AND TEST DATA

OPINION OF RESPONDENTS	NO OF RESPONDENTS	PERCENTAGE
YES	14	93
NO	1	7
TOTAL	15	100

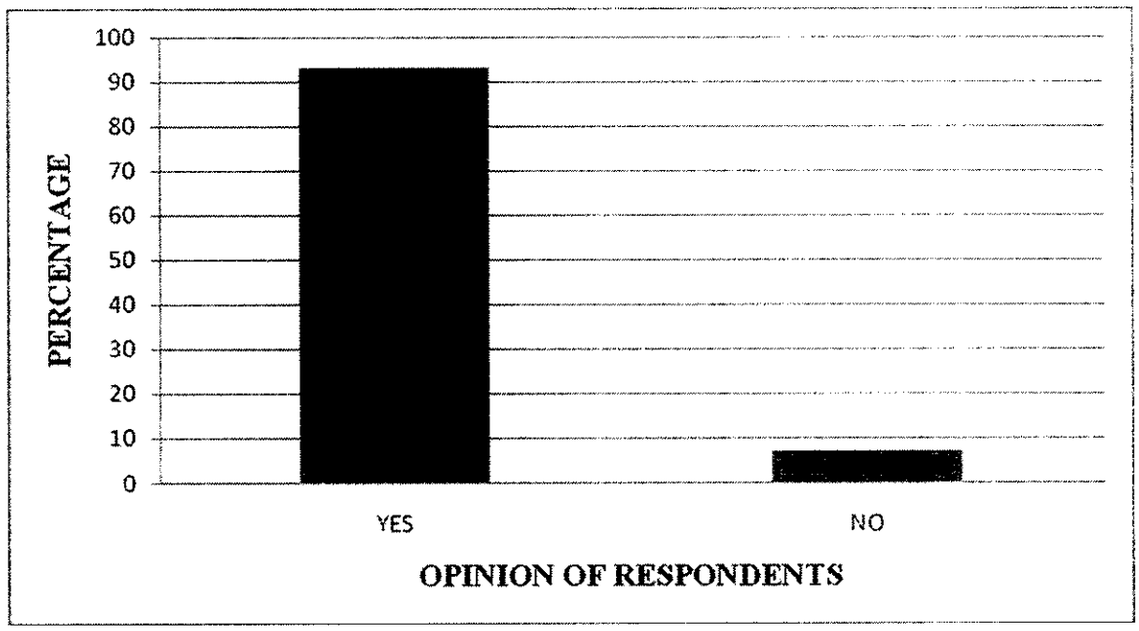


CHART No. 4.2.12 OPINION OF THE RESPONDENT REGARDING THE DESCRIPTION OF THE UNIT CASES AND TEST DATA

INTERPRETATION:

From the above table it is seen that 93% of respondents have shown a positive response with the description of the unit cases and test data but the rest 7% have shown a negative response towards it

TABLE No. 4.2.13

SATISFACTION LEVEL OF RESPONDENTS FOR INDIVIDUAL TEST ASPECTS

QUESTIONS SERIES	SATISFACTORY LEVEL (%)
Q1	60
Q2	87
Q3	93
Q4	93
Q5	87
Q6	93
Q7	80
Q8	80
Q9	73
Q10	93
Q11	60
Q12	93
OVERALL SATISFACTORY LEVEL	75

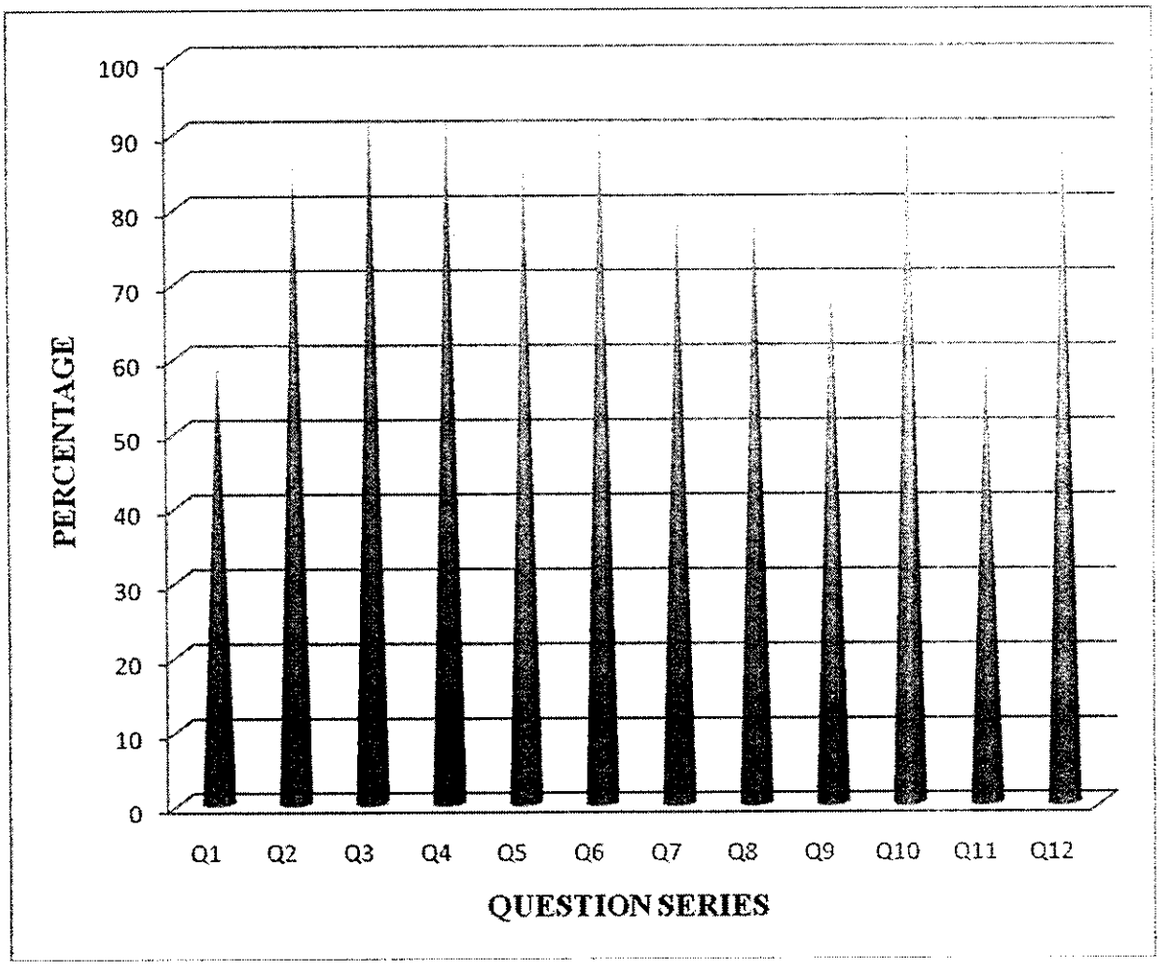


CHART No. 4.2.13 SATISFACTION LEVEL OF RESPONDENTS FOR INDIVIDUAL TEST ASPECTS

INTERPRETATION:

The overall satisfaction level of the respondents regarding the individual aspects of the testing process was found to be 75%.

4.3 TEST SUMMARY REPORT

4.3.1 DESCRIPTION OF TEST ENVIRONMENT

A single database test environment was used to test the project. The database was configured similar to the clients own set up. The database used was SQL Server 2005 on XP operating system. The process was set up on Windows XP operating system, 2.2 GHz of speed processor, Clock speed of 900MHz and 256 MB of RAM with Hard disk of 40GB.The configuration of the client workstations was 2 GHz processor and 256MB of RAM running the NT 4.0 and Windows XP operating system. The database used was SQL Server 2005.

4.3.2 TEST STRATEGY

- Test summary was developed for the Cross functional team to review the test strategy.
- A test plan was developed to specifically identify and describe how the tests cases were to be executed.
- The automation tool Quick Test Professional Version 8.2 was used for Regression testing.
- The testing mainly focused on the Projects process it self not the client; however QA did note and log numerous client issues.
- Install testing was done from the diskette to the machine.

4.3.3 SUMMARY

The testing of the Project took 32 days. The modules were divided on the basis of screens as Job Order system and Production Process for testing purpose based on the client requirement. Some of the test cases were under gone Regression Test using the functional testing tool Quick Test Professional 8.2 for adding some new functionality by the client. The testing faced many of the issues and all the issues were clearly mentioned in Test Execution Report and in Defect report. It was the first build done after the code was first sent to QA for two cycles (iterations) of testing and some of the issues for the third cycle i.e., iteration.

It was decided after the second round of testing and some of the issues for third round that QA would do sample installation testing. A finished installation testing and forward the disk and files to product management and client services. In total 18 issues were uncovered in the two testing cycles, 60% were with the process and 40% were with the client. Since the customer already has the client portion installed, it was decided that all client issues would not be postponed and will be covered with in next build release. QA was asked to provide the documentation team with instructions on how to set the second built and the client up. QA also provided the team a list of release notes in the document was decided after the second round of testing and some of the issues for third round that QA would do sample installation testing. A finished the installation testing and forward the disk and files to product management and client services.

4.3.4 METRICS

Metrics for the initial built:

Test Timeframe 04/03/2008 – 25/03/2008

Total Days 9 days, 75.5 hours of testing

Number of builds 1

Resources 1 full time, 1 part time

Total tests designed 290

Total tests executed 290

Total tests automated 10 (in 290)

Total Hours 154.5 includes planning, troubleshooting problems and actual test execution.

Test Preparation Time Total 79 Hours

Test Execution Time Total 75.5 Hours

Build1 73.5

Install Test1 2

Defect Summary

Total found 24 defects found from QA testing

Total canceled/closed 37.5 %

4.3.5 RISK ASSESSMENT

The possible risk assessments are low, medium and high. The risk assessment for this project is high

4.3.6 KNOWN DEFECT LIST

Defect Report was done in separate excel file: production Defect report.xls

5. CONCLUSIONS

CHAPTER 5
CONCLUSION

5.1 FINDINGS:

- The testing of the production process and control system unveiled the following results. Of the total 239 test cases, 221 test cases passed and 18 test cases failed.
- The job order module consists of 5.2% defects and the production process module consists of 9.2% defects.
- The production process and control system on the whole consists of 7.5% defects which do not fall into the acceptable range of 3%.
- Out the 18 defects which were uncovered during the first iteration, 13 defects were rectified by the developer's team and the remaining 5 defects were left deferred.
- The regression testing done using the automated testing tool named QTP revealed that the other features of the project remained unaltered after the inclusion of a new functionality into the customer detail form.
- The satisfaction level and dissatisfaction level of 15 members of the testing team were analyzed using peer review check lists where the members filled in their opinions regarding certain individual aspects of the testing project such as test case description, test case execution, etc.,
- The overall satisfaction level of the respondents regarding the individual aspects of the testing process was found to be 75%.

5.2 SUGGESTIONS:

The percentage of defects in the production planning and control system seems to be higher than the allowable range which denotes the inefficiency of the developer's team. Therefore the management should conduct several training programmes and seminars in order to improve the programming skills of the developers. The manual testing process is tedious and consumes lot of days. So in order to save time and money, efficient and low cost automated tools could be used. It is advisable that both the testers and the developers should be trained effectively to possess both testing and developing skills in order to overcome minor defects by themselves without depending on others.

6. REFERENCES

CHAPTER 6

REFERENCES

BOOKS

1) William E. Perry,"Effective Methods for Software Testing", Second Edition, Willey Computer Publishing Ltd.
2) Roger S. Pressman, "Software Engineering A Practitioners Approach", Sixth Edition , Tata McGraw Hill Publishing Company Ltd.
3) Richard B Chase, F Robert Jacobs, Nicholas J Aquilano, Nictin K Agarwal " Operations Management" Eleventh Edition, Tata McGraw Hill Edition.
4) Richard Fairley ," Software Engineering Concepts" , McGraw-Hill International Editions.

WEB SITES

1) www.ibm.com
2) www.ieee.org
3) http//search3.computer.search results
4) www.accessustechnologies.com
5) www.express computer onlne.com
6) www.softwaretestinghelp.com.

APPENDIX

APPENDIX

A1 SOURCE CODE SAPMPLES

```
Imports System.Data.sqlclient
Public Class Productdetail
   Inherits System.Windows.Forms.Form
Dim cn As New SqlConnection
   Protected Const cnstr As String = "workstation id=SREVER;packet
size=4096;integrated security=SSPI;initial catalog=productivity;persist security
info=False"
   Dim ds As New DataSet
   Dim sql, mt, flag As String
   Dim da As New sqlDataAdapter
   Dim cmd As New sqlCommand
   Dim mybind As BindingManagerBase
   Dim isbound As Boolean
Sub textclear()
      Dim c As Object
      For Each c In Panel1.Controls
        If TypeOf c Is TextBox Then
          c.clear()
          c.DataBindings.clear()
```

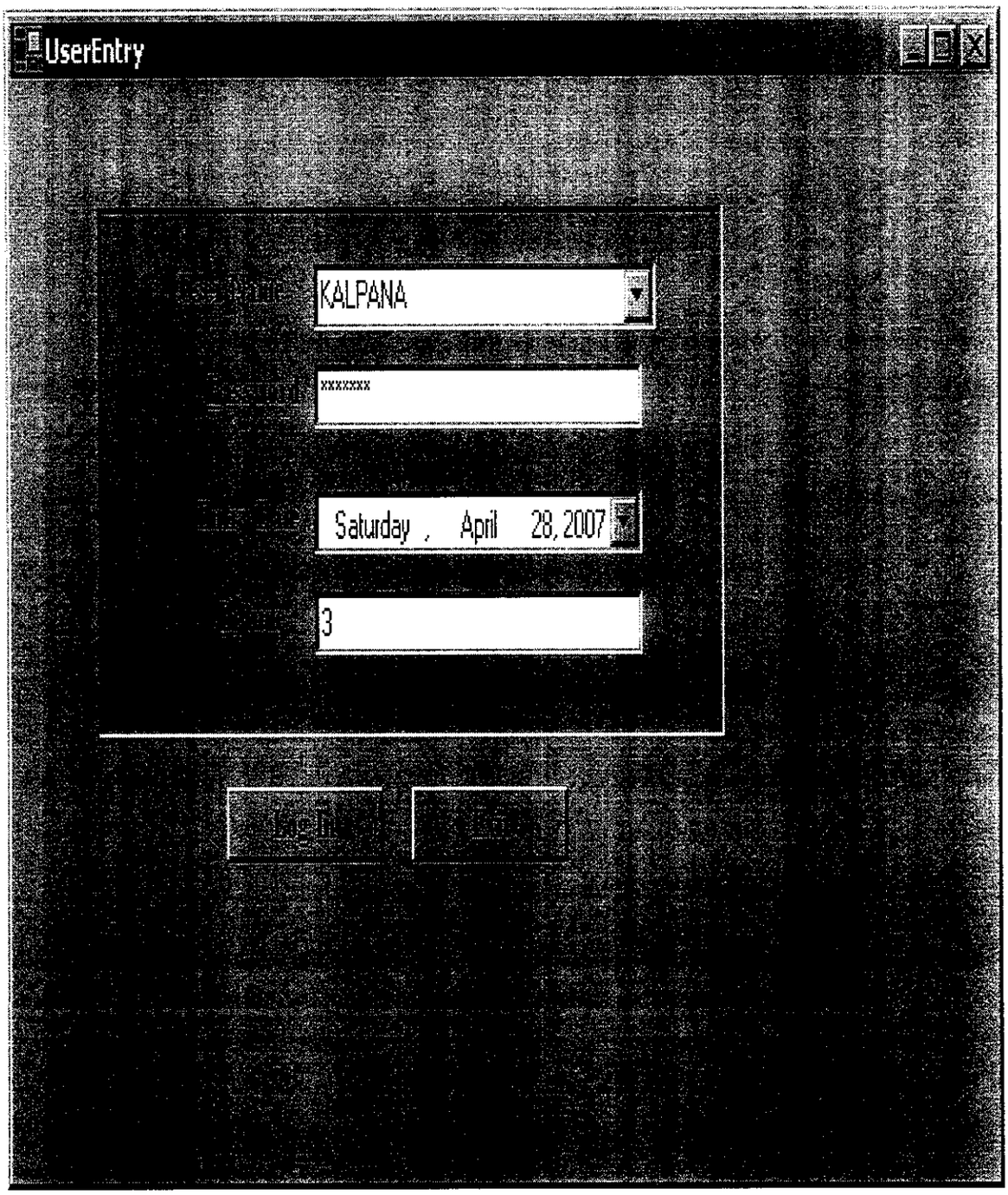
```
        End If
      Next
    End Sub
Sub sctrue()
Private Sub Toolbar1_ButtonClick(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.ToolBarButtonClickEventArgs) Handles tbrMain.ButtonClick
      If e.Button Is btnAdd Then
        textclear()
        'write the coding here autogen for record no
        flag = "new"
        sctrue()
        'spnlstatus.Text = "Add New Record"
        maxrec()
        DISPCOMBO()
        ' DateTimePicker1.Focus()
      End If
      If e.Button Is btnEdit Then
        flag = "edit"
        Panel1.Enabled = True
        sctrue()
        ' spnlstatus.Text = "Edit Record"
      End If
      If e.Button Is btnsave Then
        mt = "no"
        'emptycheck()
```

```
        Try
            If mt = "no" Then
                scfalse()
                If flag = "new" Then
                    cn.ConnectionString = cnstr
                    cn.Open()
sql = "insert into product values('" & Trim(txtrecno.Text) & "','" & Trim(txtpcode.Text) &
"','" & Trim(txtpname.Text) & "','" & Trim(txtccode.Text) & "')"
                    cmd = New SqlCommand(sql, cn)
                    cmd.ExecuteNonQuery()
                    cn.Close()
                    MsgBox("save record")
                    connectdatabase()
                    flag = ""
                End If
                '========edit to update
                If flag = "edit" Then
                    cn.ConnectionString = cnstr
                    cn.Open()
                    sql = "update product set pcode='" & Trim(txtpcode.Text) & "',pname='" &
Trim(txtpname.Text) & "',ccode='" & Trim(txtccode.Text) & "' where recno='" &
Trim(txtrecno.Text) & "'"
                    cmd = New SqlCommand(sql, cn)
            Private Sub Toolbar1_ButtonClick(ByVal sender As System.Object, ByVal e
As System.Windows.Forms.ToolBarButtonClickEventArgs) Handles tbrMain.ButtonClick
```

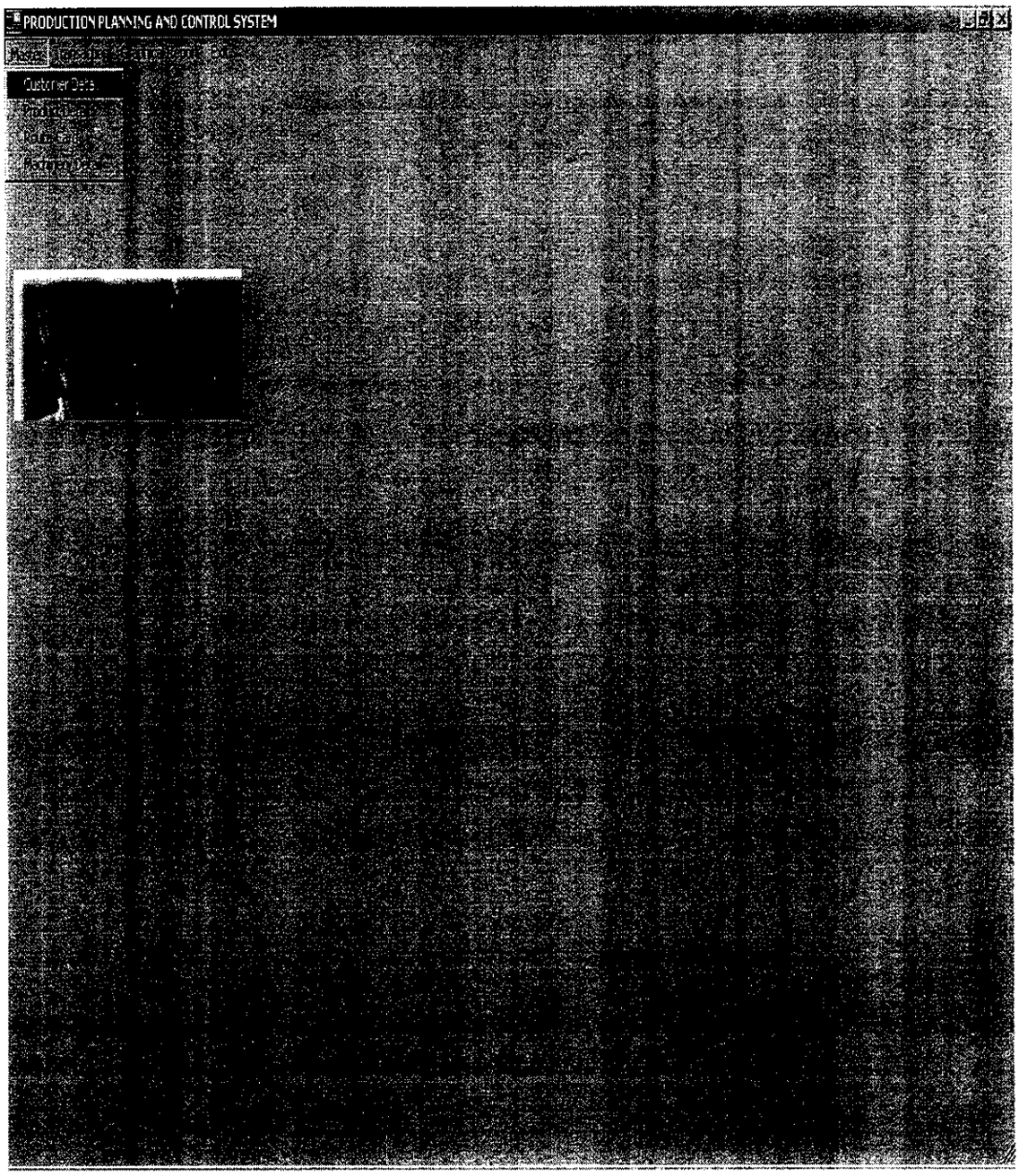
```
        If e.Button Is btnAdd Then
            textclear()
            'write the coding here autogen for record no
            flag = "new"
            sctrue()
            'spnlstatus.Text = "Add New Record"
            maxrec()
            DISPCOMBO()
            ' DateTimePicker1.Focus()
        End If
End sub
```

A2 PRODUCTION PLANNING AND CONTROL SYSTEM SNAP SHOTS

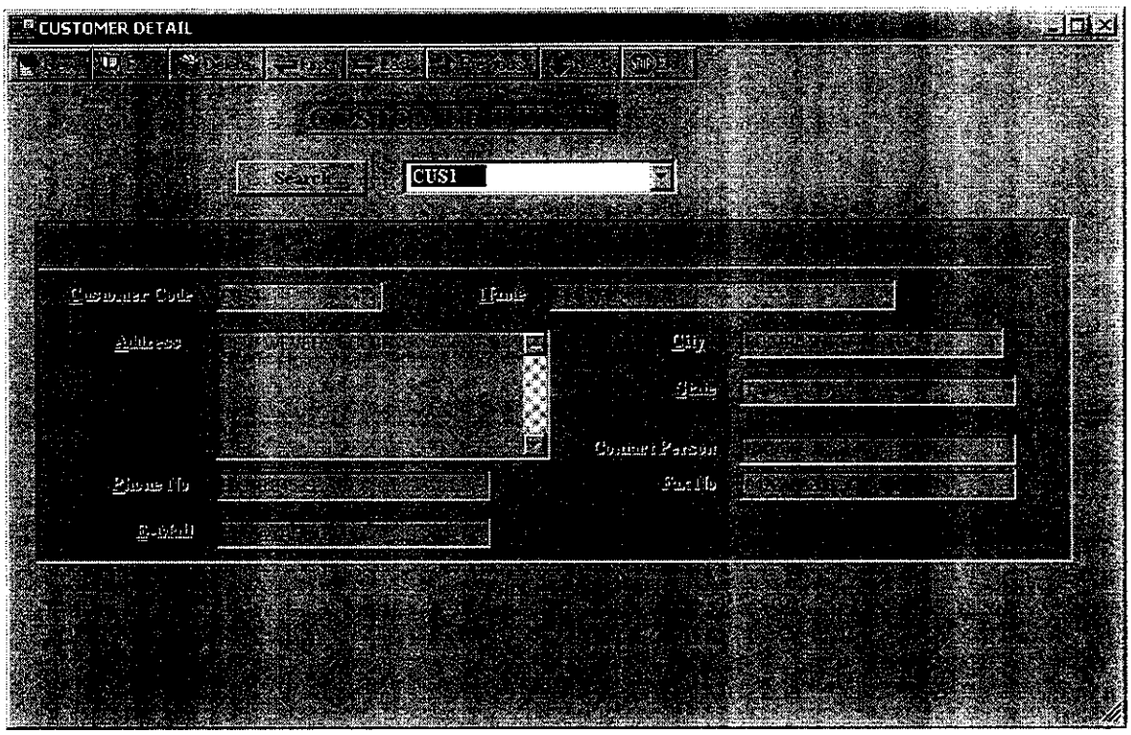
LOGIN FORM



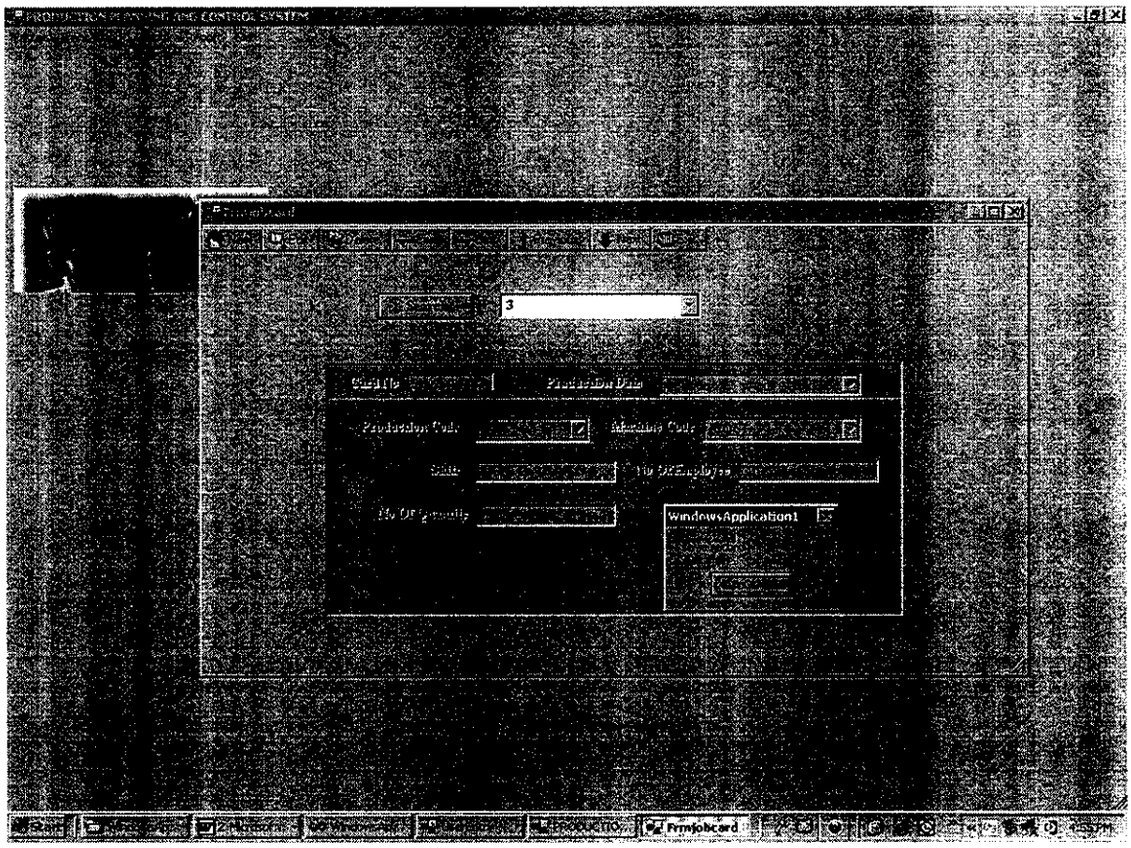
HOME PAGE



CUSTOMER DETAIL FORM



JOB CARD FORM



A3 PEER REVIEW CHECK LIST

	Peer Review Checklist-STC	

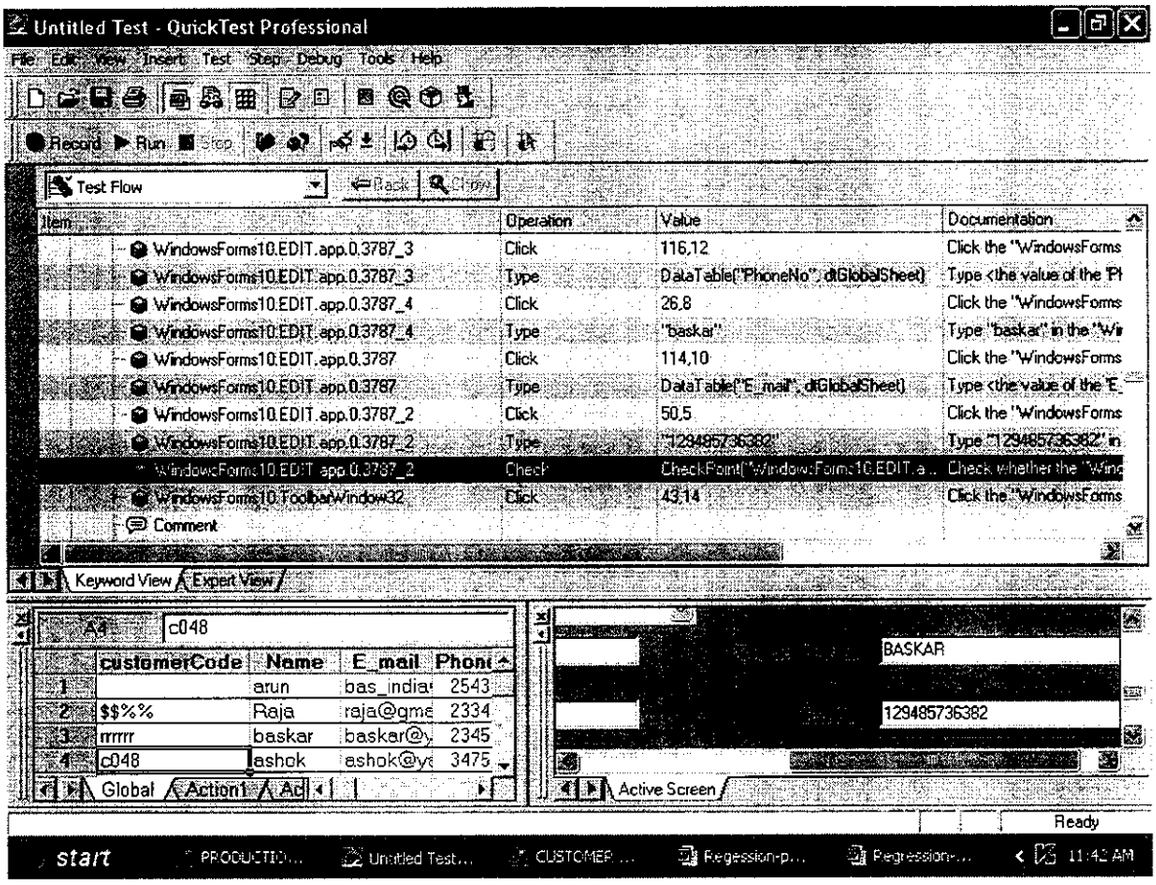
Project ID	ACT/PPCS/001
Project name	Production Planning and Control System
Review start date	07/03/2008
Review end date	12/03/2008
Document under review	Test Execution Report

SNo	Item	Yes/No/ NA	Comments
1	Is the version of the screen shot for which test cases are written specified		
3	Are the test cases accurately described?		
4	Is the functionality of the test cases clearly mentioned?		
5	Is the action of the test cases clearly mentioned?		
6	Are the expected results of test cases clearly mentioned?		
7	Are the criteria for success/failure clear and unambiguous?		
8	Do test cases lead to determination of success or failure?		
9	Is each requirement associated with this screen shot exercised by the test cases?		
10	Are the boundary conditions/ limit values checked in each of the user input values?		
11	Is expected response to each step of test case described?		
12	Are test cases prepared for all scenarios and every option in the logical loop?		
13	Do test cases demonstrate program's response to illegal and conflicting input data?		
14	Are all dependencies of test cases identified?		
15	Do expected results of the test cases agree with expected program behavior?(verify with DDD)		
16	Are test cases classified into "U" for Unit, "S" for System, "I" for Integration?		
17	Are unit cases clearly described including test data?		

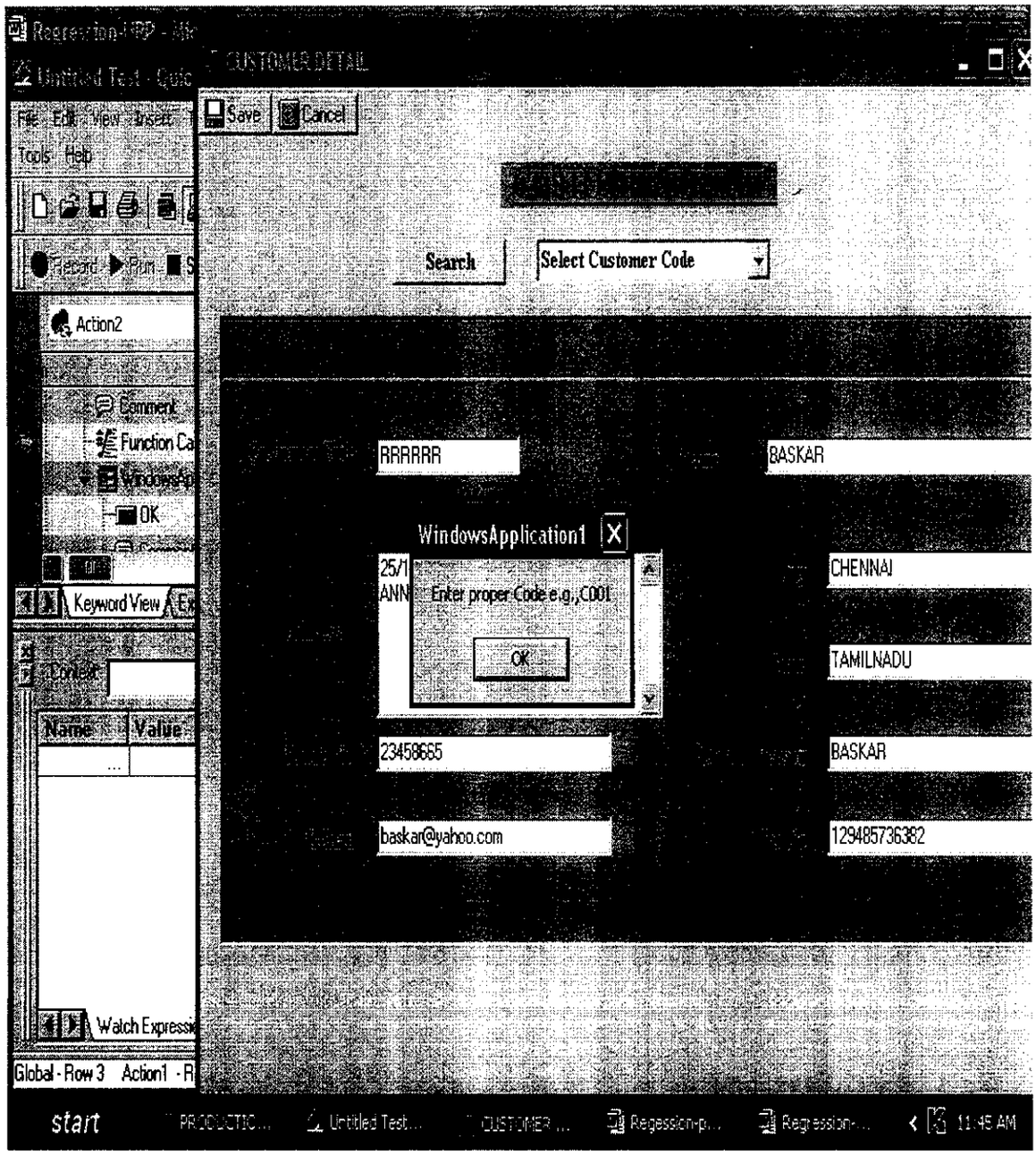
Name	Sujitha.D	
Process Role	Tester	
Signature		
Date	13/03/2008	

A 4 QTP SNAPSHOTS

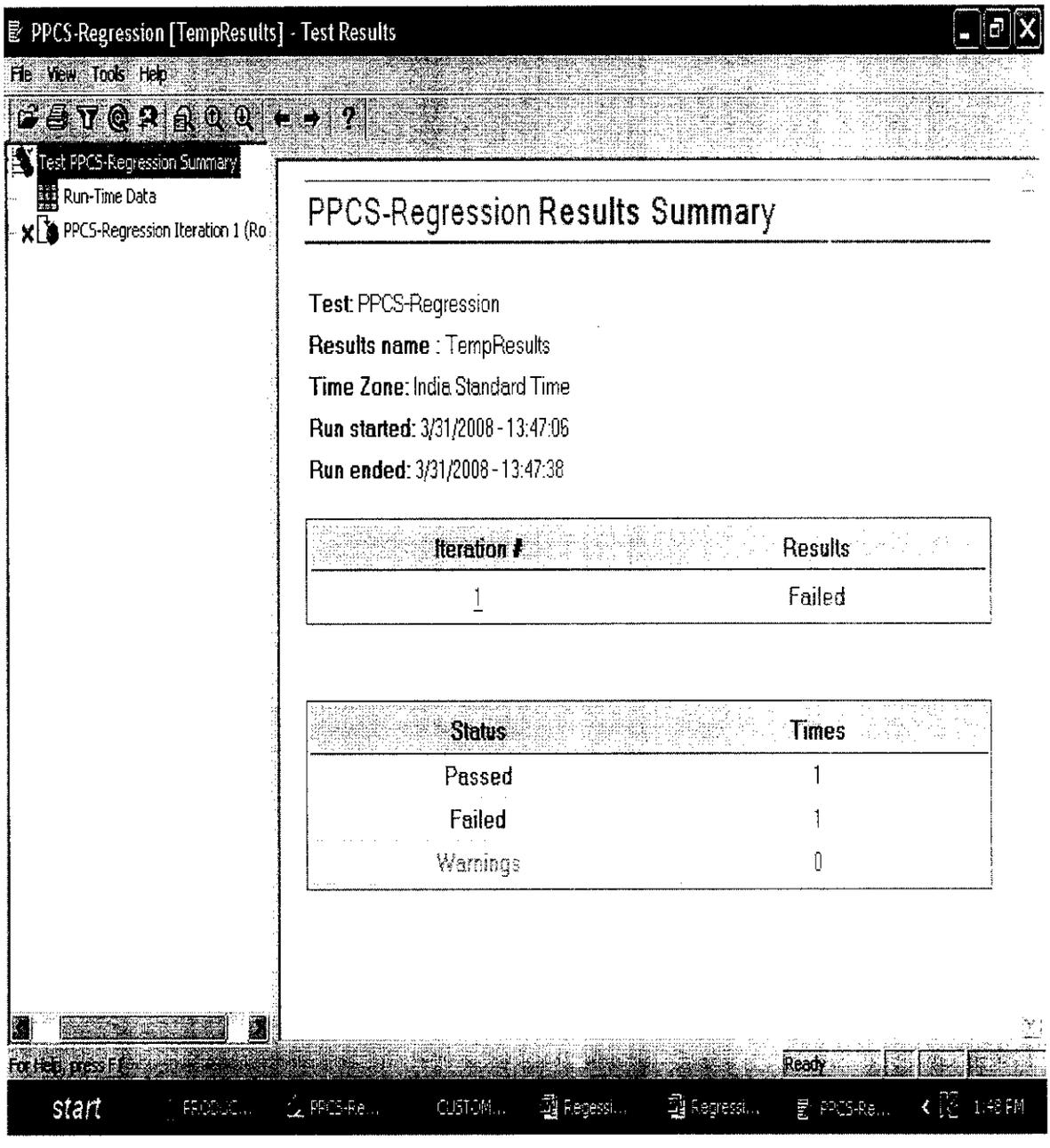
Quick Test Professional Keyword View with Active Screen and Global data sheet:



Triggering the event (Microsoft Internet Explorer Pop-up window) using Recovery Manager Scenario:

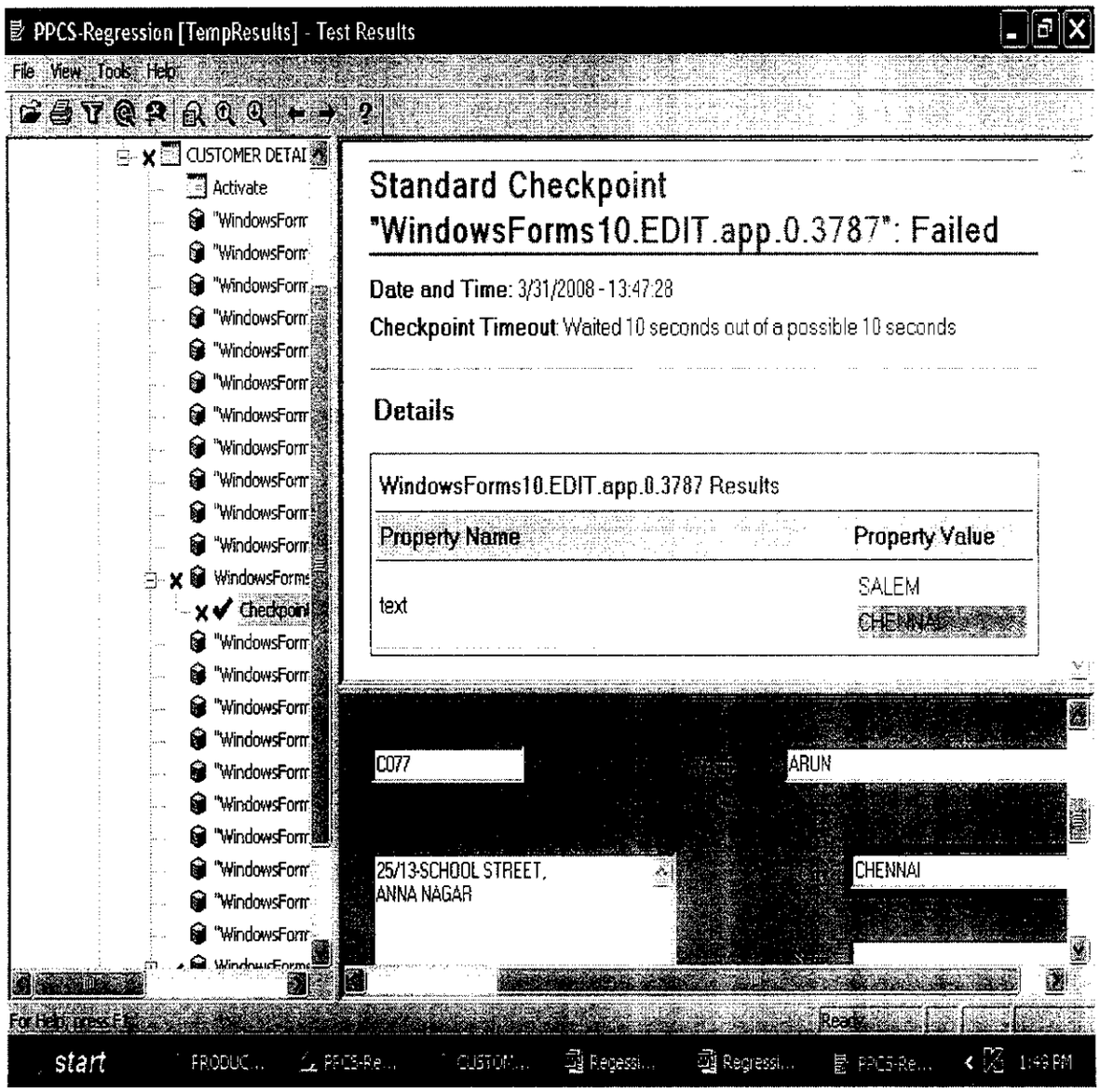


Test Result of Using Check points for the fields.



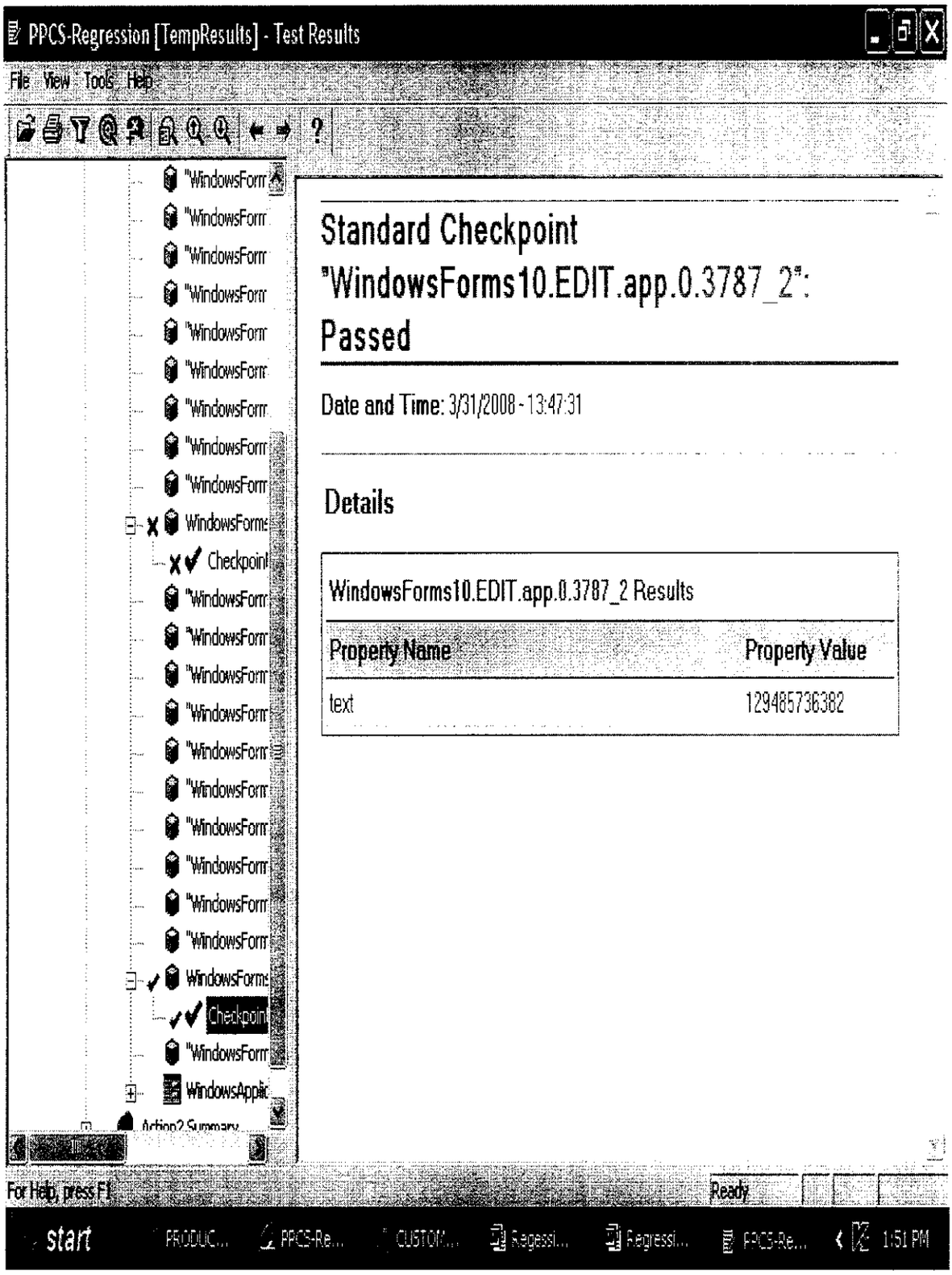
Test Results by Using Check points for the fields.

Sample Data: For City. (Negative Test Data)

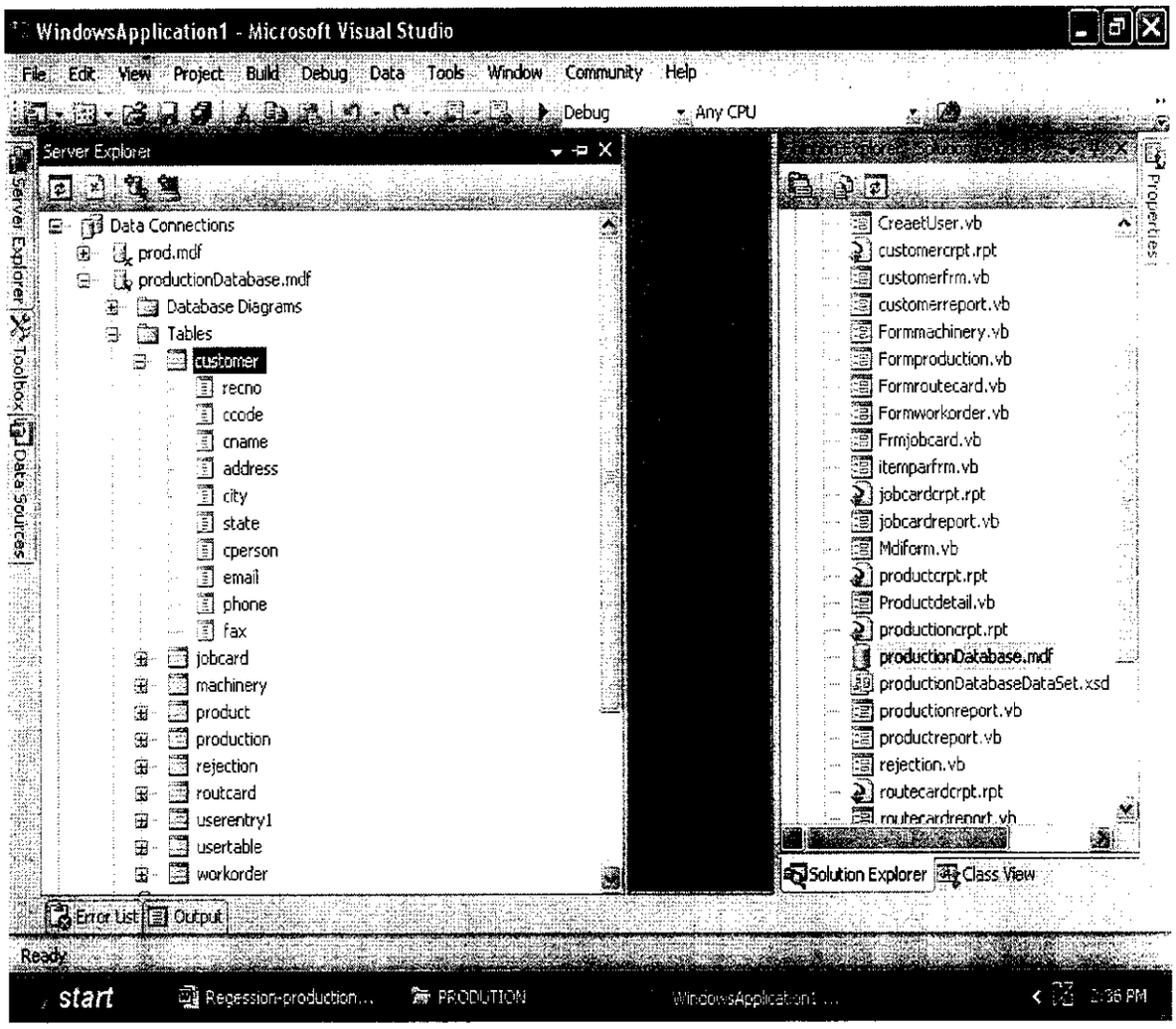


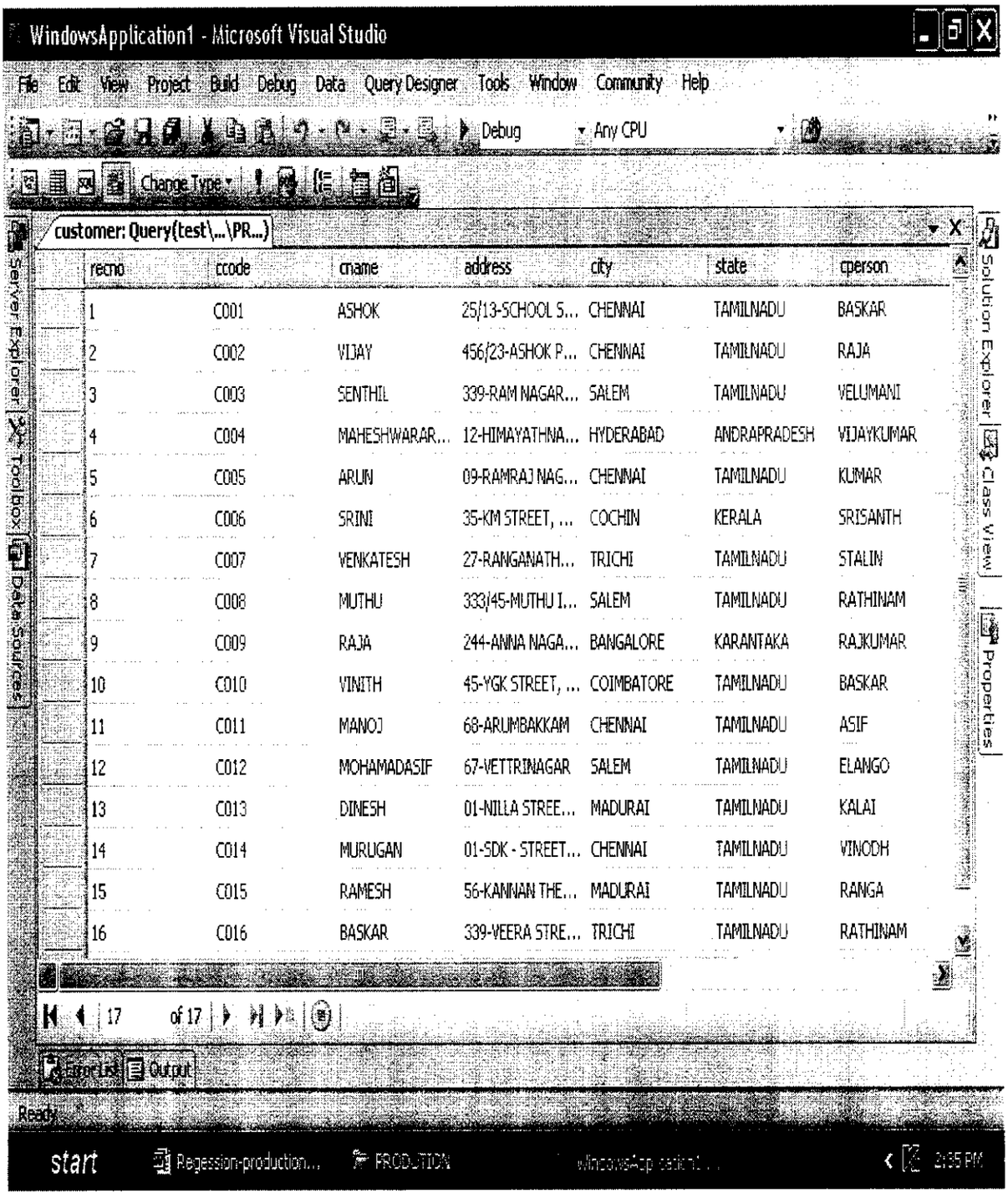
Test Result of Using Check points for the fields.

Sample Data: For Fax No. (Positive Test Data)



Accessing and Checking Databases:





recno	ccode	cname	address	city	state	cperson
1	C001	ASHOK	25/13-SCHOOL S...	CHENNAI	TAMILNADU	BASKAR
2	C002	VIJAY	456/23-ASHOK P...	CHENNAI	TAMILNADU	RAJA
3	C003	SENTHIL	339-RAM NAGAR...	SALEM	TAMILNADU	VELUMANI
4	C004	MAHESHWARAR...	12-HIMAYATHNA...	HYDERABAD	ANDRAPRADESH	VIJAYKUMAR
5	C005	ARUN	09-RAMRAJ NAG...	CHENNAI	TAMILNADU	KUMAR
6	C006	SRINI	35-KM STREET, ...	COCHIN	KERALA	SRISANTH
7	C007	VENKATESH	27-RANGANATH...	TRICHI	TAMILNADU	STALIN
8	C008	MUTHU	333/45-MUTHU I...	SALEM	TAMILNADU	RATHINAM
9	C009	RAJA	244-ANNA NAGA...	BANGALORE	KARANTAKA	RAJKUMAR
10	C010	VINITH	45-YGK STREET, ...	COIMBATORE	TAMILNADU	BASKAR
11	C011	MANOJ	68-ARUMBAKKAM	CHENNAI	TAMILNADU	ASIF
12	C012	MOHAMADASIF	67-VETTRINAGAR	SALEM	TAMILNADU	ELANGO
13	C013	DINESH	01-NILLA STREE...	MADURAI	TAMILNADU	KALAI
14	C014	MURUGAN	01-SDK - STREET...	CHENNAI	TAMILNADU	VINODH
15	C015	RAMESH	56-KANNAN THE...	MADURAI	TAMILNADU	RANGA
16	C016	BASKAR	339-VEERA STRE...	TRICHI	TAMILNADU	RATHINAM

17 of 17

Test Case and Test Execution Report

Project Name	Production Planing and Control System
Test Case Prepared By	Sujitha
Test Start Date	7/3/2008
OS / Platform	Windows Xp / DotNet
Name of the Tester	Sujitha
Test End Date	22/03/2008

Test Case ID	Functionality	Action	Expected Result	Observed / Actual Result (Iteration No. 0)	Pass / Fail (0)	Pass / Fail (1)	Pass / Fail (2)	Pass / Fail (3)	Type of Testing (IT / ST)
TC_UETY_01	Check the functionality of user loginby giving blank user name and blank password	1)don't type anything in user name 2)don't type anything in password 3)click on login button 4)click on create user button	Error message should be displayed	As Expected	Pass				
TC_UETY_02	Check the functionality of user login by giving blank user name and valid password	1)don't type anything in user name 2)enter a valid password 3)click on login button 4)click on create user button	Error message should be displayed	As Expected	Pass				
TC_UETY_03	Check the functionality of user login by giving valid user name and blank password	1)enter a valid user name 2)enter a blank password 3)click on login button 4)click on create user button	Error message should be displayed	As Expected	Pass				
TC_UETY_04	Check the functionality of user login by giving valid name and invalid password	1)enter a valid user name 2)enter a invalid password 3)click on login button 4)click on create user button	Error message should be displayed	As Expected	Pass				
TC_UETY_05	Check the functionality of user login by giving invalid user name and valid password	1)enter a invalid user name 2)enter a valid password 3)click on login button 4)click on create user button	Error message should be displayed	As Expected	Pass				
TC_UETY_06	Check the functionality of user login by giving invalid user name and invalid password	1)enter a invalid user name 2)enter a invalid password 3)click on login button 4)click on create user button	Error message should be displayed	As Expected	Pass				
TC_UETY_07	Check the functionality of user login by giving special characters	1)enter special characters(for e.g !,@,#) 2)click on login button	Error message should be displayed	As Expected	Pass				

Test Case and Test Execution Report

Project Name	Production Planning and Control system
Test Case Prepared By	Sujitha.D
Test Start Date	7/3/2008
OS / Platform	Windows Xp / DotNet
Name of the Tester	Sujitha.D
Test End Date	23/03/2008

Test Case ID	**Functionality**	**Action**	**Expected Result**	Observed / Actual Result	Iteration No. 0 **Pass / Fail**	Iteration No. 1 Pass / Fail	Iteration No. 2 **Pass / Fail**	Iteration No. 3 **Pass / Fail**	**Type of Testing (IT / ST)**
TC_CUST_01	Check the functionality of new button	click on the new button	1)All the fields should be enabled 2)The customer code field should be focused	As Expected	**Pass**				ST
TC_CUST_02	check the fieldlevel validation of customer code by leaving blank	1)don't enter anything in the text box field 2)Click on save button	Error message should be displayed	As Expected	**Pass**				ST
TC_CUST_03	check the fieldlevel validation of customer code by entering characters	1)enter characters in the text box field 2)Click on save button	Error message should be displayed	As Expected	**Pass**				ST
TC_CUST_04	check the fieldlevel validation of customer code by entering numerics	1)enter numerics in the text box field 2)Click on save button	Error message should be displayed	As Expected	**Pass**				ST
TC_CUST_05	check the fieldlevel validation of customer code by entering special characters	1)enter special characters in the text box field 2)Click on save button	Error message should be displayed	As Expected	**Pass**				ST
TC_CUST_06a	check the field level validation of customer code by entering alpha numeric	1)enter alpha numeric characters **(say C001)** 2)Click on save button	It should accept and no error message should be displayed	As Expected	**Pass**				ST
TC_CUST_06b	Check the fuctionality of customer code for duplicate values	Enter alpha numeric values that already entered and saved	Error message should be displayed	As Expected	**Pass**				
TC_CUST_07	check the fieldlevel validation of address by leaving blank	1)don't enter anything in the multi line text box field 2)Click on save button	Error message should be displayed	As Expected	**Pass**				ST
TC_CUST_08	check the fieldlevel validation of address by without leaving blank	1)enter anything in the multi line text box field 2)Click on save button	Should accept and error message should not be displayed	As Expected	**Pass**				ST
TC_CUST_09	check the field level validation of phone no by leaving blank	1)Don't enter anything in the text box field 2)Click on save button	Error message should be displayed	As Expected	**Pass**				ST
TC_CUST_10	check the fieldlevel validation of phone no by entering characters	1)enter characters in the text box field 2)Click on save button	Error message should be displayed	As Expected	**Pass**				ST

DEFECT TRACKING REPORT

S No.	Defect ID	Test Case ID	Module Name	Functionality/Summary	Defect Description	Opened Date	Tester	Severity	Status	Fixed Date	Closed Date	Reopened Date	Comments
1	D1_01	TC_MDI_02_05	JobOrdSys	productivity form does not exist	1) Enter valid user name (As a admin or user) 2) Enter Valid password 3) Click on "Login" button 4) Click on the "transaction" tab list on MDI Form. 5) Click on the "productivity" Observed Behaviour: The productivity form does'nt displayed. Expected Behaviour: The Productivity form should be displayed	7/3/2008	Sujitha	S4	Open				
2	D1_02	TC_RUT_06	JobOrdSys	No product code gets displayed when clicking on the drop down list box	1) Enter valid user name. 2) Enter Valid password. 3) Click on "Login" button. 4) Click on the "master" tab list on MDI form. 5) Click on the "route card". 7)click on the product code drop down list box on Route card Form. Observed Behaviour: The product code should get displayed when clicking on the drop down list box. Expected Behaviour: No product code gets displayed when clicking on the drop down list box.	13/03/2008	Sujitha	S3	Closed	15/03/2008	17/03/2008		The product code should be available to select
3	D1_03	TC_RUT_13	JobOrdSys	Didn't displays product code	1) Enter valid user name. 2) Enter Valid password. 3) Click on "Login" button. 4) Click on the "master" tab list on MDI form. 5) Click on the "route card". 6)click on the first button on the route card form. Observed Behaviour: Didn't displays product code from the record Expected Behaviour: First record details should be displayed	14/03/2008	Sujitha	S3	Re-Open	15-Mar-08		21/03/2008	
4	D1_04	TC_RUT_14	JobOrdSys	Didn't displays product code	1) Enter valid user name. 2) Enter Valid password. 3) Click on "Login" button. 4) Click on the "master" tab list on MDI form. 5) Click on the "route card". 6)click on the last button on the route card form. Observed Behaviour: Didn't displays product code from the record Expected Behaviour: last record details should be displayed	14/03/2008	Sujitha	S3	Re-Open	17-Mar-08		21/03/2008	