

ONLINE SURVEY SYSTEM – TECHNOLOGY AT THE DOORSTEP

Suvayan Sengupta

IT Analyst

Business Role: Software Architect / SME

Follow me on KNOME: <https://knome.ultimatix.net/u/suvayan.sengupta>

Tata Consultancy Services

Building 1A, Ecospace

Plot - IIF/12, New Town, Rajarhat,

Kolkata - 700156, West Bengal

India

Buzz:- 82538

Cell:- +91 8013777267

Mailto: suvayan.sengupta@tcs.com

Website: <http://www.tcs.com>

Abstract

In citation to the earlier issue, Modern Trend Analysis in Software Product Development the Online Survey system is getting pioneered. It is used in questionnaires, quizzing and survey in which any user can add his question and also gives the response of any polls. The following gives a vivid description of its operational details, description, objectives, testing procedures with real time samples clubbed with its future scope. A CMMI Level 5 certified organization has been explored for this project work.

Keywords: Software product, Modern Technology, Cloud Design, Online System, Survey

Introduction

In the previous paper, **Modern Trend Analysis in Software Product Development: A path model for predicting Technology (Harvest, Volume-2/2016)**, we already discussed the explosion of internet enabling instant accessibility of any kind of information to anyone giving 5th gear to Modern project management with virtual machineries offering actual technology. On that note, to give an extra mileage we introduce the Survey System. It is frequently used in survey-systems, and questionnaires. **Computer Survey System** is software that can work as real SURVEY SYSTEM. It also provides the facilities like question designing and editing, printing of results, and analysis of results etc. The system allows a multiple choice question to be posed online. The poll participant is invited to select the most appropriate option and to record their reasoning in a text entry box. On submission of their vote the participant is presented with a results page which graphically illustrates the distribution of previous responses to the poll and the reasoning of previous voters grouped by the option they selected. It has following facilities like:

A System, like an online system, is used in questionnaires, quizzing and survey in which any user can add his question and also gives the response of any polls.

We also can send the mails to the users who are interested in our website.

We can also include the e-library facility in system through which users can also upload and download the books.

Users can also add the add sense which is the very interesting facility in the system.

We also enhance the popularity of the system through combined with any other social networking sites.

Graphical representation of result of polls.

This software reduces the cost of survey sheet by using SURVEY DEPARTMENT to find growth of country.

Survey is done on the basis of many categories like age group of users, type of users.

2.0 System Study

2.1 Description of Existing System

Basically, every organization has many parts, and everybody should maintain some basic information. Therefore every organization should have at least some process of working to track and serve their users. Suppose that a government organization want to perform survey on different peoples that lives in country then it will check all records manually. Survey System can increase profitability for survey field by reducing current operating costs. It can save time in test or manage the records, and save the cost of records, make the polling in analytical manner, to handle bulk amount of data, and provide facilities of inviting groups.

There are few survey system but those have many problems contained in it like:

These systems can't send email to interact with users.

We can't add ad sense in these systems which are very interesting feature of our system.

These system's don't have facility of e-library means we can't download books from these systems.

These systems don't have any graphical facility to represent result of the survey.

2.2 Description of Current System

In order to reduce the time of process and getting information we try to make it easy and proper way so that user get the information in easy manner. We have developed the system which work's in efficient and faster manner and works according to the need like:

We developed the system in PHP which very faster technology to developed web applications.

We also include facilities of email; add sense, graphical representation of the result of the survey.

We have facility to do survey on the basis of different categories.

We reduce the unnecessary burden of the database as far as possible.

There is uniform layout and design of screens across the system.

The system provides all the necessary security functions to prevent from misuse and unauthorized access the system.

2.3 Objective of Project

The main objectives of project are:

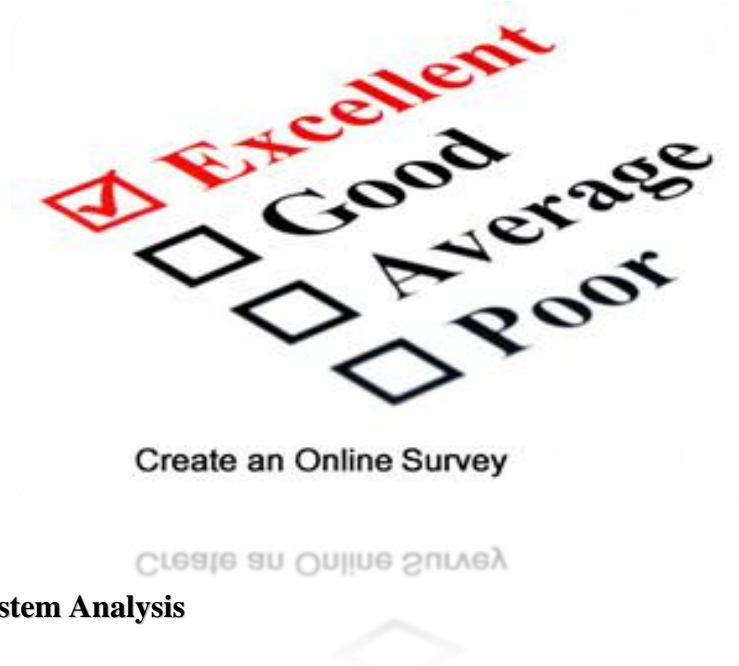
Provide a system through which users online will do survey of their products, also get opinion of people online about what they think about the particular problem.

Advertising of the Survey System so that it used by many people's online.

Less time consuming and reduce the burden of database.

To get the information in easy and efficient way.

To make work simple and user friendly



3.1 Analysis: System Analysis

3.1.1 Problem Definition

Maintaining the records manually, creates lot of confusion and cause mismanagement.

The workload is biased to a single worker as the records are maintained manually.

The status of all the records is not updated regularly.

Manual data recording system is time consuming and takes lot of efforts to maintain it.

Major problem through manual entry is that it doesn't provide the technical specification of the particular product.

Problem faced by employee is that they have to go for each place and collects the information about polling system which takes lots of time and laborious also.

3.1.2 Feasibility Study

It is a procedure that identifies, describes, and evaluates candidate systems and selects the best systems for the job. The feasibility study of the project is essentially made. It was feasible. So the project is taken up. User needs/features of the product are captured at the beginning of the project. An estimation of the size of the projects in fiction points is done. In some cases, it may be difficult to make accurate estimates. The number of iterations the project may go through may also be not known. It is important that the customer knows the limitations under which the estimates has been made.

3.1.3 Technical Feasibility

This study checks the technical aspects of system. It also checks the available software technology and manpower. In this project PHP Framework 6.0 is used. PHP is Object Oriented nature, Multiplatform application development, significant improvements in code reuse, Code specialization, Automatic resource management, Security, Ease of deployment and Administration. This software is simple to use and manage.

3.1.4 Behavioral Feasibility

The system follows behavioral feasibility because of its friendliness in nature. Anyone can operate easily, for this we have developed user interface and user-friendly system. An old person who works on his own style cannot shift on another style. So no one wants a change,

that's why project should be feasible. This project is windows based and we provide graphical user interface which makes it easy to use. No technical training is required to use this software.

3.1.5 Economic Feasibility

This study is mainly concerned with cost-benefit analysis i.e. how much money the user is investing in any system and how much he is getting as a benefit in output. Our project is Economical Feasible because anyone uses this software would need not to buy this machine. Our hardware requirement is not too expensive. So as compare to benefit the project is economically feasible.

3.2 Analysis: Requirement Analysis

3.2.1 Requirement Specification

A requirement specification is a structured document, which sets out the system services in detail. This document, which is sometimes called a functional specification, should be precise. It may serve as a contract between the system buyer and the software developer.

3.2.2 Goals

The goal of our project is to make evaluation process of Objective questions easier and simpler for the organizations who are conducting these types of survey.



And also reduce the time and cost needed by the Polling System for this evaluation process.

3.2.3 Scope



The scope of our project **SURVEY SYSTEM** software is as follows:

Save the time for manual evaluation of survey and poll.

Make easy for user by using questions of different categories in the place of special questions sheet.

Provide facility for preparing Model Question and Answer.

Make Results in more graphical manner.

3.2.4 Functional & Non-Functional Requirements

Show frames for displaying the commands.
Use Mouse for handling the flow of processing.
Show Width of the frame within the screen range.
Use Keyboard for entering data.

3.2.5 Assumption & Dependencies

Only properly polled answers by user will be accepted for correct evaluation.
Users must be registered first for any process.
No two options will be accepted and entry will be discarded for similar question.
Categories Details should be filled by Admin.
Only Admin can provide Right Answers for the Questions.
Right Answers cannot be modifying once they have stored in database.

3.2.6 Validation Criteria

The very first validation is on our categories sections which should be according to the predefined format.
The graph of answers also checked before evaluation of answers.
Polling should be registered before Preparing Question.
User Authentication is must in our project for the login as a Admin.
No two options will be accepted and entry will be discarded for similar question.
Questions Details should be filled by Admin only.
Only Admin can provide Right Answers for the Questions.

3.3 Analysis: Technology Used

Software Specifications

Operating System : Windows 2000/XP/Windows 7/Win10
Design Tool : IBM Lotus Notes 7.0.1
Server : IBM Domino Server.

Hardware Specifications

Hard Drive Capacity : 80GB
Floppy Disk Drive : 1.44MB
RAM : 512Mb or higher
CPU : Intel Pentium IV
Printer : Dot Matrix Printer, DeskJet Printer
Monitor : VGA/SVGA Monitor or higher
A Network Interface

3.4 Analysis: Prpgramming Tool

3.6.1 IBM Lotus Notes

IBM Lotus Notes and Domino 7 Clients are available defines the Lotus Notes Database, outlines how to create a Lotus Notes database, and describes how to set up the database for application design. IBM/Lotus defines the Lotus Notes client as “the leading integrated email and collaborative software for the Internet.”

Since Release 6.0, Lotus Notes has comprised three individual software Clients, each client geared toward a specific function and user type:

Lotus Notes 7 Client (for the standard user)

Domino Administrator 7

Domino Designer 7

3.6.2 Domino Designer Client

The Domino Designer is an integrated development environment enabling Notes Developers and Web designers to create, manage, and deploy secure interactive Notes client/server and Web applications. However, in addition to the three clients offered by Lotus, you may be designing applications for other client types:

iNotes Web Access client: A Web client that delivers Domino messaging, collaboration, and PIM (Personal Information Management) capabilities to Web browsers

iNotes for Microsoft Outlook: A collaborative server solution that delivers the messaging, calendar, scheduling, and personal information management (PIM) services of Lotus Domino technology to Outlook 2000 and 2002 clients.

Other POP/IMAP clients: Other third-party email clients utilizing POP (Post Office Protocol) and IMAP (Internet Messaging Access Protocol)

Mobile clients: Clients that include PDA's, Internet enabled phones, and so on

The Domino Designer 7 client enables developers to design applications for Notes clients, Web browser clients, and Mobile clients (a feature enhanced with Lotus Domino 7) with the same rapid application development (RAD) Tool, which uses a consistent programming environment for all support languages. In addition, we can easily integrate third-party application development Tools, and Domino 7 supports Web-based Distributed Authoring and Versioning (WebDAV), enabling users to access and edit Domino design elements without using the Lotus Domino Designer clients (assuming that users are using a editor that supports WebDAV.

3.6.3 Lotus Notes Database

The Database object is an essential element in creating Domino Web applications and/or native Lotus Notes client/server applications. A Domino database is an object store that contains data, logic, and design elements for the application. A Domino database is not synonymous with an application. On the contrary, most sophisticated applications comprise several Domino databases. There is no inherent difference between the structural format and definition of databases for the Web, native clients/servers, or mobile applications. A Domino application enables users to share, collect, track, and organize information.

To open a Notes database for the first time from the Notes client, follow these steps:

Select **File, Database, Open** from the pull-down menu.

Enter the server in the server field. (Either select the server from the drop-down menu or type the name of the server if it doesn't show up.)

Select the database from the database list, or type the name of the database file in the filename field (be sure to include the **.nsf**[Notes Storage Facility], **.ntf**[Notes Template Facility], or **.ns5**extension [Notes database that retains the R5 on disk structure]).

Alternatively, select the Browse button to locate the database from a local drive.

After a database has been opened or bookmarked, you can open the database by either clicking on the bookmark or double-clicking the database icon from the workspace.

Alternatively, one may also open a database from the command line. From the command prompt (for example, MS-DOS), change to the Notes root directory.

To open a database stored locally on the current machine, enter designer Notes: [//Database_Replica_ID](#).

To open a database stored on a server, enter designer Notes: [//Server_Name/Database_Replica_ID](#).

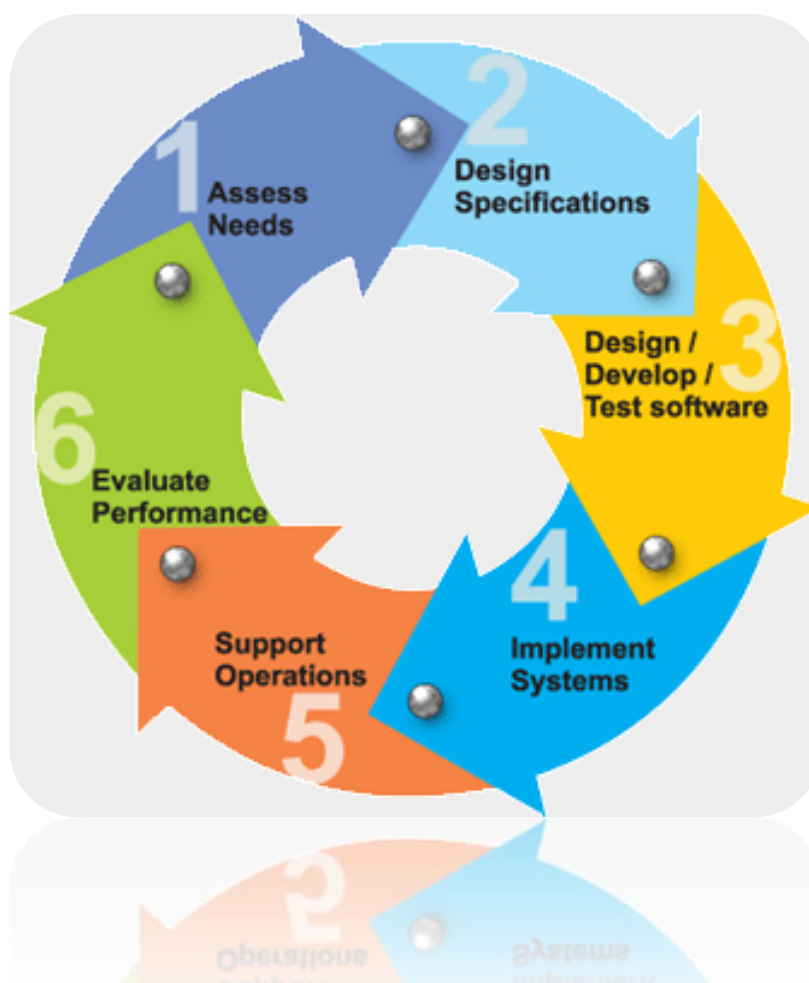
In addition, one may also open a specific document from the command line by entering designer Notes: [//Server_Name/Database_Replica_ID/Note_ID](#). When entering the Replica

ID of the database, do not include the colons. When entering the Note ID of the document, enter the characters following the NT characters displayed in the beginning of the Note ID. The ID used when the Lotus Notes client is launched must have Designer access to the database and the Domino Designer must have been opened before.

The Domino 6 Server and Notes 6 client use a different On Disk Structure (ODS) than R5 (and all previous releases). The ODS of a database does not replicate and has no effect on which client version can access a server version. The ODS only determines how the server of client writes data to the physical drive. The Information tab of the database properties displays the ODS version. Release 3.x had an ODS of 17, Release 6.x had an ODS of 20, Release 6.x had an ODS of 41, and Notes and Domino 6 has an ODS of 43.

3.5 Analysis: Software Process Model

A software development process is a structure imposed on the development of a software product. A process model for software engineering is chosen based on the nature of the project and application, the methods and tools to be used, and the controls and deliverables that are required. A common process framework is established by defining a small number of framework activities that are applicable to all software projects, regardless of their size or complexity. A number of tasks set each a collection of software engineering work tasks, project milestones, and a common process framework is established by defining a small number of framework activities that are applicable to all software projects, regardless of their size or complexity.



3.6.1 Model Used
This business process - Survey Management System uses the Water Fall Model.

3.6.2 Advantages and Disadvantages

The advantages of this model are:-

Enforced discipline through documents

Testing is inherent in every phase continuously as well as at end of phases.

The disadvantages of this model are:-

Real projects rarely follow the sequential flow that the model proposes. Although the linear model can accommodate iteration, it does so indirectly. As a result, changes can cause confusion as the project team proceeds. It is often difficult for the customer to state all requirements explicitly. The linear sequential model requires this and has difficulty accommodating the natural uncertainty that exists at the beginning of many projects.

The customer must have patience. A working version of the program(s) will not be available until late in the project time-span. A major blunder, if under- tested until the working program is reviewed, can be disastrous.

3.6.3 Reason behind usage of the model

The waterfall model is widely used by such large software development houses as those employed by the U.S. Department of Defense and NASA, and for many large government projects. Those who use such methods do not always formally distinguish between the pure waterfall model and the various modified waterfall models, so it can be difficult to discern exactly which models are being used and to what extent.

4.0 System Planning

Planning future endeavors, though difficult is important in managing operations. Planning cuts across all phases of the system life cycle. It is the first step in developing and managing systems. The objectives are to map out the developments of major systems and reduce the number of small, isolated systems to be developed and maintained. Planning for any system must be done within the framework of the organization's overall plan.

It may be viewed from two dimensions:

The time horizon dimension specifies whether it is short range, medium term or long range.

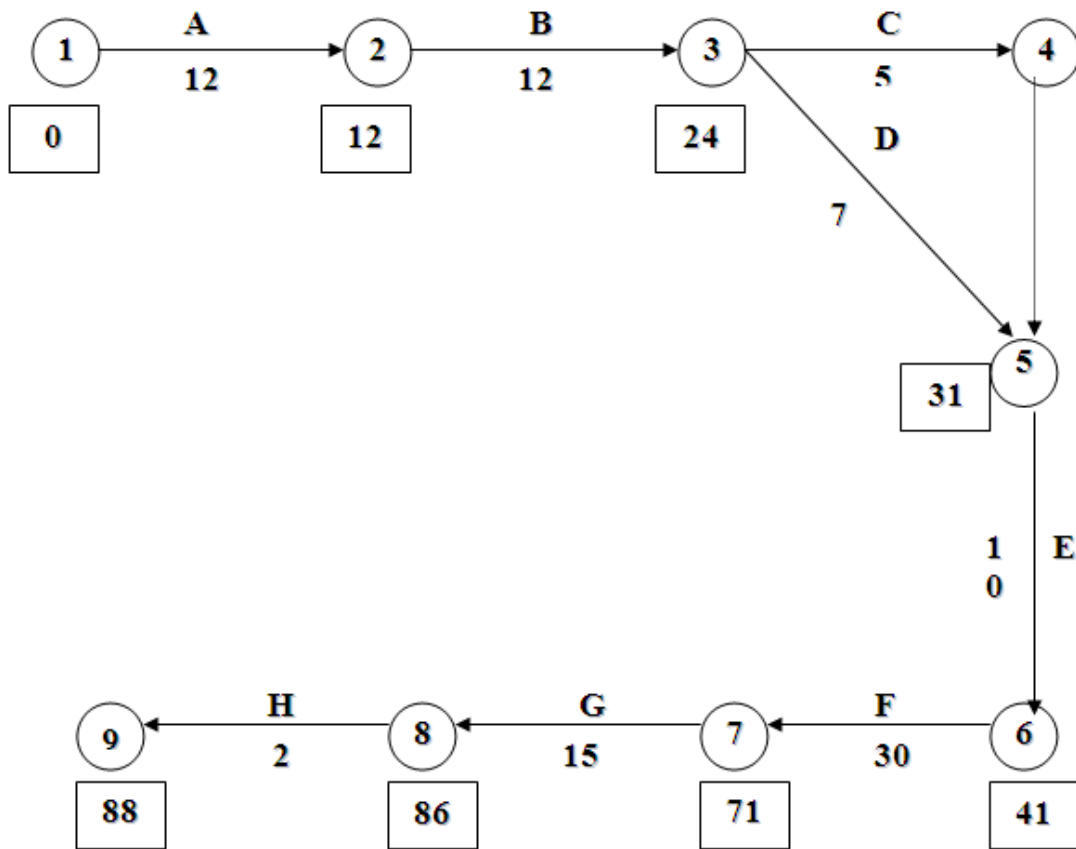
The focus dimension, which actually specifies all the tasks that, needs to be undertaken.

4.1 Pert Chart

Program Evaluation and Review Technique (Pert) makes use of tasks represented as arrows and circles represent the beginning or completion of a task. The chart uses these paths and events to show the interrelationships of project activities.

TASKS NAME	TASK ALIAS	TIME NEEDED (DAYS)
SYSTEM STUDY	A	12
INFORMATION GATHERING	B	12
FEASIBILITY STUDY	C	5
DATABASE DESIGN	D	7
FORM DESIGN	E	10
CODING	F	30
TESTING WITH SAMPLE DATA	G	15
USER/OPERATIONAL MANUAL	H	2

Pert Chart:-



TOTAL 88 DAYS

CRITICAL PATH: A → B → D → E → F → G → H

5.0 Design

Design is the bridge between system analysis and system implementation. System design is the process of planning a new system or to replace the existing system. Simply, system design is like the blueprint for building, it specifies all the features that are to be in the finished product.

System design phase follows system analysis phase. Design is concerned with identifying functions, data streams among those functions, maintaining a record of the design decisions and providing a blueprint the implementation phase.

5.0.1 Fundamental Design Concept

Some of the essential fundamental concepts involved in the design of application software are:-

Abstraction is used to construct solutions to problem without having to take account of the intricate details of the various component sub problems. Abstraction allows system designer to make step-wise refinement, which at each stage of the design may hide, unnecessary details associated with representation or implementation from the surrounding environment.

Modularity is concerned with decomposing of main module into well-defined manageable units with well-defined interfaces among the units. This enhances design clarity, which in turn eases implementation, Debugging, Testing, Documenting and Maintenance of the software product. Modularity viewed in this sense is a vital tool in the construction of large software projects.

Verification is fundamental concept in software design. A design is verifiable if it can be demonstrated that the design will result in implementation that satisfies the customer's requirements. Verification is of two types namely,

Verification that the software requirements analysis satisfies the customer's needs.

Verification that the design satisfies the requirement analysis.

5.0.2 Basic Quality Factors

Some of the important factors of quality that are to be considered in the design of application software are:-

Reliability: The software should behave strictly according to the original specification and should function smoothly under normal conditions.

Extensibility: The software should be capable of adapting easily to changes in the specification.

Reusability: The software should be developed using a modular approach, which permits modules to be reused by other application, if possible.

Preliminary Design: Preliminary design is basically concerned with deriving an overall picture of the system. Deriving entire system into modules and sub-modules while keeping Cohesion and Coupling factors in mind. Tools, which assist in preliminary design process, are Data Flow Diagrams.

Code design: The purpose of code is to facilitate the identification and retrieval for items of information. A code is an ordered collection of symbols designed to provide unique identification of an entity or attribute. To achieve unique identification there must be only one place where the identified entity or the attribute can be entered in the code; conversely there must be a place in the code for everything that is to be identified. This mutually exclusive feature must be built into any coding system.

Input Design: Input design is a part of overall system design, which requires very careful attention. The main objectives of input design are:-

To produce a cost-effective method of input.

To achieve the highest possible level of accuracy.

To ensure that the input is acceptable to and understood by the user staff.

Output Design: Outputs from computer systems are required primarily to communicate the results of processing to users. They are also to provide a permanent hard copy of these results for later consultation.

5.1 DESIGN: Entity Relationship Diagram (ER Diagram)

5.1.1 Entity-Relationship Diagram

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationship between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities.

Diamonds are normally used to represent relationships and ovals are used to represent attributes. It was introduced by Chen in 1976.

Entity-Relationship diagram describes overall logical structure of a database graphically.

These E-R diagrams are very simple and clear. It consists following major components:

Rectangles : It represents entity set.

Ellipse : It represents attributes.

Diamond : It represents relationship set.

Lines : It links attributes to entity sets and entity sets to relationship sets.

5.1.2 Type of Relationship

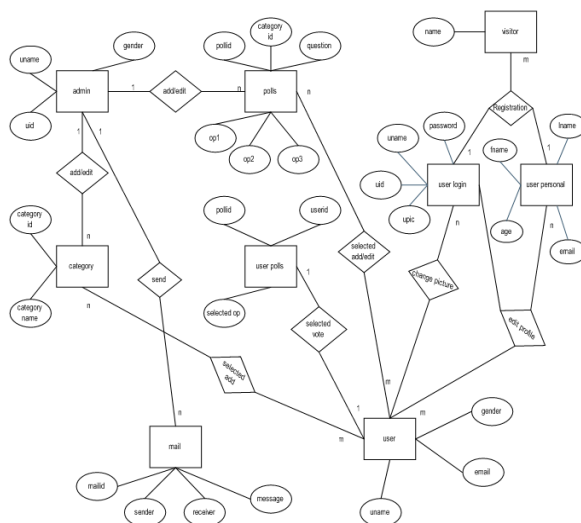
Three main types of relationship can exist between entities.

ONE-TO-MANY: It exists when one occurrence of the second entity can be associated with any one occurrence of the first entity.

MANY-TO-ONE: It exists when many occurrence of the first entity can be related to one occurrence of the second entity can be associated with any one occurrence of the first entity.

MANY-TO-MANY: It exists when one instance of the first entity can be related to many instances of the second entity and one instance of the second entity can be related to many instance of the first entity.

5.1.3 ER Diagram of Online Survey System



5.2 DESIGN: Database Design

S No.	Table Name	Attribute name	Type	Extra
1.	User_Personal	Uid	Int(5)	Auto increment
		Fname	Varchar(20)	
		Lname	Varchar(20)	
		Gender	Smallint(1)	
		Email	Varchar(60)	
		Age	Smallint(1)	
		Income	Smallint(1)	
		Race	Smallint(1)	
		Affiliation	Smallint(1)	
2.	User_Login	Uid	Int(5)	Auto increment
		Uname	Varchar(20)	
		Password	Varchar(20)	
		Verify	Smallint(2)	Default(1)
		Upic	Varchar(40)	Default (default.jpg)
		Utype	Smallint(2)	Default (0)
3.	Poll	pid	Int(5)	Auto increment
		cid	Int(3)	
		Question	Varchar(50)	
		Op1	Varchar(40)	
		Op2	Varchar(40)	
		Op3	Varchar(40)	
		Visiable	Smallint	Default (1)
		Polltype	Smallint(1)	Default (0)
		Createdby	Varchar(20)	
4.	User_poll	Index	Int(8)	
		Uid	Int(5)	
		Pid	Int(3)	
		Selectedop	Smallint(2)	
5.	User_type	Utype	Smallint(2)	

		Typename	Varchar(20)	
6.	Category	cid	Int(3)	Auto incremented
		cname	Varchar(40)	
7.	User_verify	Uid	Int(5)	
		Value	Varchar(100)	

5.3 Design: User Interface

Screen Name	Snapshot
Home Screen	

Project Details	<div style="text-align: center; background-color: #333; color: #fff; padding: 5px;">CUSTOMER SURVEY</div> <hr/> <div style="background-color: #f0f0f0; padding: 10px;"> <p>Project Details</p> <p>Select Quarterly/Half- Yearly/Annually <input type="text" value="Quarterly"/> Survey for the Financial Year : <input type="text" value="2012-13"/></p> <p>Select your Quarter : <input type="text" value="Apr-Jun"/></p> <p>PROJECT: ITS KOLKATA Tag : Y MODULE : M&S CUSTOMER DEPARTMENT : TUBES DIVISION STATUS : Survey Done CUSTOMER NAME : BHARAT CHANDRA LAHA/CALCUTTA/TATA STEEL LIMITED/IN@TATA STEEL EXTERNAL NAME : COMPANY NAME : P K Mishra</p> <p style="text-align: center;">MAIL TO CUSTOMER</p> </div>																								
Survey Details (1)	<div style="background-color: #f0f0f0; padding: 10px;"> <p style="text-align: center; background-color: #333; color: #fff; padding: 5px;">Survey Form</p> <p style="color: #d9534f; font-weight: bold;">Kindly indicate your rating on the following dimensions :</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 80%;"></th> <th style="width: 20%; text-align: center; color: #333; font-weight: bold;">Remarks / Area for Improvement</th> </tr> </thead> <tbody> <tr> <td colspan="2">1. Service Level</td> </tr> <tr> <td>System availability <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> <tr> <td>On-line response time <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> <tr> <td>Batch job turn around time <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> <tr> <td>Adherence to monthly schedule <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> <tr> <td colspan="2">2. Support Provide By Help Desk</td> </tr> <tr> <td>Availability of contact person <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> <tr> <td>Attitude of contact person <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> <tr> <td colspan="2">3. Responsiveness of support Agencies</td> </tr> <tr> <td>Hardware support <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> <tr> <td>Software support <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor</td> <td style="text-align: center;">Ok</td> </tr> </tbody> </table> </div>		Remarks / Area for Improvement	1. Service Level		System availability <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok	On-line response time <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok	Batch job turn around time <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok	Adherence to monthly schedule <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok	2. Support Provide By Help Desk		Availability of contact person <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok	Attitude of contact person <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok	3. Responsiveness of support Agencies		Hardware support <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok	Software support <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok
	Remarks / Area for Improvement																								
1. Service Level																									
System availability <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								
On-line response time <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								
Batch job turn around time <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								
Adherence to monthly schedule <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								
2. Support Provide By Help Desk																									
Availability of contact person <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								
Attitude of contact person <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								
3. Responsiveness of support Agencies																									
Hardware support <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								
Software support <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor	Ok																								

<p>Survey Details (2)</p>	<p>4. Responsiveness to user for changes in application (wrt changes / problems etc)</p> <p>Turn around time <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor Ok</p> <p>User contact and communication <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor Ok</p> <p>5. Any other dimension of product / service that you would like to rate</p> <hr/> <p>6. Overall satisfaction level <input checked="" type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Fair <input type="radio"/> Poor Ok</p> <p>*Please enter reason for your satisfaction level (This is for us to improve on our services and products)</p> <p>7. Key dimension for Dissatisfaction</p> <table border="1"> <tr> <td>Interface Problem</td> <td>Application</td> <td>Remarks</td> <td>Ratings</td> </tr> <tr> <td>Response Time</td> <td>Application</td> <td>Remarks</td> <td>Ratings</td> </tr> <tr> <td>LAN Connectivity</td> <td>Application</td> <td>Remarks</td> <td>Ratings</td> </tr> <tr> <td>Coverage of Reports</td> <td>Application</td> <td>Remarks</td> <td>Ratings</td> </tr> <tr> <td>Training</td> <td>Application</td> <td>Remarks</td> <td>Ratings</td> </tr> </table> <p>8. What are your suggestions for improvig the computerized systems already implemented ?</p> <p>a) IT System :</p> <p>b) Business Processes :</p>	Interface Problem	Application	Remarks	Ratings	Response Time	Application	Remarks	Ratings	LAN Connectivity	Application	Remarks	Ratings	Coverage of Reports	Application	Remarks	Ratings	Training	Application	Remarks	Ratings
Interface Problem	Application	Remarks	Ratings																		
Response Time	Application	Remarks	Ratings																		
LAN Connectivity	Application	Remarks	Ratings																		
Coverage of Reports	Application	Remarks	Ratings																		
Training	Application	Remarks	Ratings																		
<p>View Reports</p>	<table border="1"> <tr> <th>CURRENT REPORTS</th> <th>ARCHIVED REPORTS</th> </tr> <tr> <td> <input checked="" type="checkbox"/> COMPLETED SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE <input checked="" type="checkbox"/> DETAILS OF CSI <input checked="" type="checkbox"/> ALL SURVEYS <input checked="" type="checkbox"/> DEPARTMENT WISE <input checked="" type="checkbox"/> CUSTOMER PROJECT WISE <input checked="" type="checkbox"/> PROJECT CUSTOMER WISE <input checked="" type="checkbox"/> STATUS WISE <input checked="" type="checkbox"/> PENDING SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE </td> <td> <input checked="" type="checkbox"/> COMPLETED SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE <input checked="" type="checkbox"/> DETAILS OF CSI <input checked="" type="checkbox"/> ALL SURVEYS <input checked="" type="checkbox"/> DEPARTMENT WISE <input checked="" type="checkbox"/> CUSTOMER PROJECT WISE <input checked="" type="checkbox"/> PROJECT CUSTOMER WISE <input checked="" type="checkbox"/> STATUS WISE <input checked="" type="checkbox"/> PENDING SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE </td> </tr> </table>	CURRENT REPORTS	ARCHIVED REPORTS	<input checked="" type="checkbox"/> COMPLETED SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE <input checked="" type="checkbox"/> DETAILS OF CSI <input checked="" type="checkbox"/> ALL SURVEYS <input checked="" type="checkbox"/> DEPARTMENT WISE <input checked="" type="checkbox"/> CUSTOMER PROJECT WISE <input checked="" type="checkbox"/> PROJECT CUSTOMER WISE <input checked="" type="checkbox"/> STATUS WISE <input checked="" type="checkbox"/> PENDING SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE	<input checked="" type="checkbox"/> COMPLETED SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE <input checked="" type="checkbox"/> DETAILS OF CSI <input checked="" type="checkbox"/> ALL SURVEYS <input checked="" type="checkbox"/> DEPARTMENT WISE <input checked="" type="checkbox"/> CUSTOMER PROJECT WISE <input checked="" type="checkbox"/> PROJECT CUSTOMER WISE <input checked="" type="checkbox"/> STATUS WISE <input checked="" type="checkbox"/> PENDING SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE																
CURRENT REPORTS	ARCHIVED REPORTS																				
<input checked="" type="checkbox"/> COMPLETED SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE <input checked="" type="checkbox"/> DETAILS OF CSI <input checked="" type="checkbox"/> ALL SURVEYS <input checked="" type="checkbox"/> DEPARTMENT WISE <input checked="" type="checkbox"/> CUSTOMER PROJECT WISE <input checked="" type="checkbox"/> PROJECT CUSTOMER WISE <input checked="" type="checkbox"/> STATUS WISE <input checked="" type="checkbox"/> PENDING SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE	<input checked="" type="checkbox"/> COMPLETED SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE <input checked="" type="checkbox"/> DETAILS OF CSI <input checked="" type="checkbox"/> ALL SURVEYS <input checked="" type="checkbox"/> DEPARTMENT WISE <input checked="" type="checkbox"/> CUSTOMER PROJECT WISE <input checked="" type="checkbox"/> PROJECT CUSTOMER WISE <input checked="" type="checkbox"/> STATUS WISE <input checked="" type="checkbox"/> PENDING SURVEYS <input checked="" type="checkbox"/> PROJECT WISE <input checked="" type="checkbox"/> YEAR WISE																				

6.0 Testing

6.1. Testing Objective

As a guiding stone to **Modern Trend Analysis in Software Product Development**, the objective of software testing is to check whether a software has errors or not and if it has then remove those errors. Before releasing a software last step or process is software testing and it is performed at the end of software development life cycle. Many people think that software testing is different from quality assurance. As quality assurance is processes which take place throughout the software development life cycle (SDLC). We perform software testing to give quality products to the users and for maintaining quality of software, software testing is essential. In different situations, different strategies are used.

For example, if you want to test a web based software or a website then you have to use load testing or stress testing. If you want to test software and you know about its internal functionalities and source code then you will use White Box testing technique.

However, if you want to test software but you don't know about its source code and internal function then now your strategy will be different. As this time depending on the situation you

will perform Black Box testing. So what I want to say that which strategy we will use to test software depends on the conditions.

The Major Objectives of Software Testing:

Uncover as many as errors (or bugs) as possible in a given timeline.

Demonstrate a given software product matching its requirement specifications.

Validate the quality of software testing using the minimum cost and efforts.

Generate high quality test cases, perform effective tests, and issue correct and helpful problem reports.

6.2 Testing Principles

Software testing is an extremely creative and intellectually challenging task. When testing follows the principles given below, the creative element of test design and execution rivals any of the preceding software development steps.

Testing must be done by an independent party. Testing should not be performed by the person or team that developed the software since they tend to defend the correctness of the program.

Assign best personnel to the task. Because testing requires high creativity and responsibility only the best personnel must be assigned to design, implement, and analyze test cases, test data and test results.

Testing should not be planned under the tacit assumption that no errors will be found.

Test for invalid and unexpected input conditions as well as valid conditions.

The program should generate correct messages when an invalid test is encountered and should generate correct results when the test is valid.

The probability of the existence of more errors in a module or group of modules is directly proportional to the number of errors already found.

Testing is the process of executing software with the intent of finding errors.

Keep software static during test. The program must not be modified during the implementation of the set of designed test cases.

Document test cases and test results.

Provide expected test results if possible. A necessary part of test documentation is the specification of expected results, even if providing such results is impractical.

6.3 Testing Methods

Software testing methods are traditionally divided into black box testing and white box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

Specification-based testing: Specification-based testing aims to test the functionality of software according to the applicable requirements. Thus, the tester inputs data into, 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), either "is" or "is not" the same as the expected value specified in the test case. Specification-based testing is necessary, but it is insufficient to guard against certain risks.

White box testing: White box testing, by contrast to black box testing, is when the tester has access to the internal data structures and algorithms (and the code that implement these). The following types of white box testing exist:

API testing: Testing of the application using Public and Private APIs.

Code coverage: creating tests to satisfy some criteria of code coverage.

Fault injection methods

Mutation testing methods

Static testing: White box testing includes all static testing.

Black box testing: Black box testing treats the software as a "black box," without any knowledge of internal implementation. Black box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing.

6.4 Scope of testing

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and code generation. Testing is a process of executing a program with the intent of finding an error once source code has been generated software must be tested to uncover (and correct) as many error as possible before delivery to the customer.

During testing, the program to be tested is executed with a set of test cases, and the output of the program for the test cases is evaluation to determine if the program is performing as it is expecting error is the testing fundamental and is defined as the difference between actual and ideal. Testing is usually defined upon to detect the fault in the coding phases. The main goal is to design a series of test cases that have a high likelihood of finding errors.

6.5 Testing Methodology

During testing different level of testing are used which performs different task.

Unit Testing:

Unit testing focuses on verification efforts on the smallest unit of software design the software component or module. In unit testing individual components are tested to ensure that they operate correctly. Each component is tested independently, without other system components. In the similar way, all the forms are first tested individually for different test data.

Module Testing: A module is a collection of dependent components. A module encapsulates related components so can be tested without system modules. In the similar way, we tested for taxi management module, customer management module, employee management module and booking module.

Sub-System Testing: This phase involves testing collection of modules which have been integrated into sub-system may be independently designed and implemented. The sub-system test process concentrated on the decision of the interface errors by rigorously exercising the interfaces. In the similar way, we combined the three modules and performed the testing of sub-system. We tested that whether each module is working properly or not.

System Testing: The sub-systems are integrated to make up the entire system. The testing process is concerned with finding errors which result from unanticipated interaction between sub-system and system components. It is also concerned with validating that the system meets its functional and non-factional requirements. Similarly, we integrated sub-system consisting of customer, employee and taxi with various functions. After testing all the sub-systems we tested each and every link provided on each page. Thus the finally we tested the complete site.

7.0 System Implementation And Evaluation

Implementation is used here to mean the process of converting a new or a revised system design into an operational one. Implementation of a new system to replace an existing one is

usually difficult. If not properly planned, there can be many problems. This phase is less creative than system design and is primarily concerned with user training, site preparation and file conversion.

The type of implementation that is incorporated in the development of the project is implementation of a modified application to replace an existing one, using the same computer. This type of conversion is relatively easy to handle.

This project deals with the design of the web based application, the back-end and application software, the front-end. The back-end contains Tables and the front-end contains Forms.

During the software-testing phase each module of software is thoroughly tested for bugs and for accuracy of output. The system developed is very user-friendly and the detailed documentation is also given to the user as online help wherever necessary. The implementation phase normally ends with the formal test involving all the components.

Hence the design of the entire system is user-friendly and simple the implementation has been quite easy.

8.0 Limitation and Future Scope

Limitations:

As with every project, this approach has also got some of the restriction, which should be followed when it used. These restrictions are listed below:

User must have a valid email-id.

User can vote any poll only one time.

Only registered user can add and edit polls and categories.

Hence finally we can conclude that if handled with care this project will lead to an efficient approach in answer checking.

Future Scope:

Survey System is an effort to make a system through which we can find out the result of survey online among various category of user's easily and instantly. In the future we can enhance our system in the following ways:

We add our system in various social networking sites in which many people communicate with each other and also we get the opinion of those people what they think about the particular problem.

Used in future as online market profit analysis.

In future we can add a chatting facility, and a quiz game, etc.

9.0 Conclusion

Our aim was to develop an online computerized system giving more edge to the Modern Trend Analysis in Software Product Development discussed in earlier issue, which would be able to fulfill all the basic requirement of user (owner/user's) with a more digital oriented approach and we are successful to achieve most of the features, which has been proposed by us, but certain limitations are also there. Our software is user friendly. It can be easily understood by any person who is able to operate computer. Database is used to store information so that it will be useful in future.

On the basis of testing, we find that our approach is working properly and correctly without any error in all cases. This approach satisfies all validity constraints, which we applied.

10.0 Bibliography

Books:

IBM Domino Developer Works Red Book Edition

IBM Lotus Notes 7.0.1 Bible 3rd Edition

Sengupta, S. 2016. Modern Trend Analysis in Software Product Development: A Path Model for Predicting Technology 2:43-83

Websites:

http://en.wikipedia.org/wiki/IBM_Lotus_Notes

<http://www.ibm.com/developerworks/lotus/documentation/dominodesigner/>

<http://www.ibm.com/developerworks/lotus/documentation/dominodesigner/65x.html>