

**SARDAR PATEL COLLEGE OF ENGINEERING-BAKROL**

Date:31/12/18

**PRACTICAL: 1**

**Aim:Study the software Microsoft Visio.**

What is Microsoft Visio:-

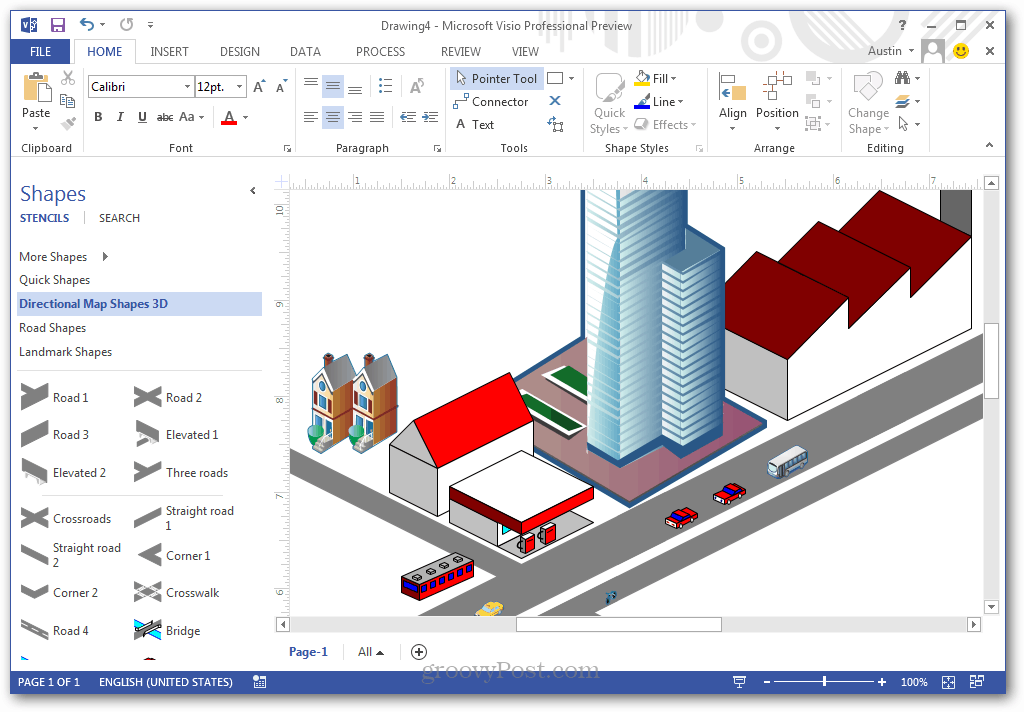
* Visio is a program that falls under the Microsoft Office Suite of products. It is used for many things that utilize layouts, diagrams, and charts. The graphics that are used in Visio are standard images utilized by flowcharts, decision diagrams, playbooks, and even network diagramming. It even has enough diagrams to make planning a room layout for a conference accurate and simple.
* Microsoft Visio is software for drawing a variety of diagrams. These include flowcharts, org charts, building plans, floor plans, data flow diagrams, process flow diagrams, business process modeling, swimlane diagrams, 3D maps, and many more. It’s a Microsoft product, sold as an addition to MS Office. Visio 2016, the latest version, comes in three editions: Visio Standard, Visio Professional, and Visio Pro for Office 365.
* you have several steps that can be followed to help determine a cause of computer problem. For example, if the computer is not turning on, but the power light is on, it's likely you need to turn on the monitor. Of course, this is just an example and not the full set of steps required to [troubleshoot](https://www.computerhope.com/jargon/t/troushoo.htm) a computer.

Features:-

* A few new features have been added such as one-step connectivity with Excel data, information rights management (IRM) protection for Visio files, modernized shapes for office layout, detailed shapes for site plans, updated shapes for floor plans, modern shapes for home plans.
* Microsoft made Visio 2013 for Windows available in two editions: Standard and Professional. The Standard and Professional editions share the same interface, but the Professional edition has additional templates for more advanced diagrams and layouts, as well as capabilities intended to make it easy for users to connect their diagrams to data sources and to display their data graphically. The Professional edition features three additional diagram types, as well as intelligent rules, validation, and subprocess (diagram breakdown).

Uses:-

* Microsoft Visio can be used to create simple or complicated diagrams. It offers a wide variety of built-in shapes, objects, and stencils to work with. You can also make your own shapes and import them if you’re willing to do all that extra work. The driving idea behind Visio is to make diagramming as easy as possible for the user.
* Microsoft Visio is a diagramming tool that can be used to visually communicate technical as well as non-technical representations of ideas, processes, concepts, structures, layouts, software models, blueprints, etc. Microsoft Visio eliminates the laborious process of creating diagrams by providing the tools to create complex diagrams in a user friendly manner. Whether you want to create an office layout, a basic flowchart or an organizational structure, you can do it with minimal effort using Microsoft Visio. The latest version is Visio 2010 which simplify complexity with dynamic, data-driven visuals and new ways to share on the Web.
* Visio can be used to create 3D map diagrams, though the built-in tools for this are limited. It works well for simple maps that you might print on a brochure or campus directory.



* Another thing Visio can do is pull in live information from an external source, such as an Excel sheet or Access database. This makes diagrams functional and current. The most recent example I’ve seen involved using Visio to monitor network status across a localized broadband system.
* Visio comes packed with a lot of built-in shapes you can add to your creation.

**Conclusion:**

In above practical we learn about What is Microsoft Visio, features of Microsoft Visio and uses of Microsoft Visio.



**SARDAR PATEL COLLEGE OF ENGINEERING-BAKROL**

Date:07/01/19

**PRACTICAL: 2**

**Aim: Study the complete Software Development life cycle and analyze various activities conducted as a part of various phases. For each SDLC phase,identify the objective and summaries outcomes.**

What is SDLC?

* SDLC is the acronym of Software Development Life Cycle.
* It is the process used for building a high quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.
* Basically, it is a framework defining tasks performed at each step in the software development process.

Graphical representation of SDLC.

* **Requirement Gathering:** Requirement gathering is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry.
* This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.
* Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.
* **Analysis:** After the customer provides requirements for the product, the project manager and members of the project team begin to analyze the requirements. The business managers analyze each requirement to ensure the requirement can be included in the software without causing breaks or problems with system functionality.
* **Design:** The system and software design documents are prepared as per the requirement specification document. This helps define overall system architecture. This design phase serves as input for the next phase of the model.

It has two steps:

**HLD** – High Level Design – It gives the architecture of the software product to be developed and is done by architects and senior developers  
**LLD** – Low Level Design – It is done by senior developers. It describes how each and every feature in the product should work and how every component should work.

* **Implementation:** During implementation, the project team creates the actual product. Product implementation can be an exciting phase for the customer, because their idea for the project becomes something tangible. Project developers begin building and coding the software.
* **Testing:** This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.
* **Maintenance:** The maintenance phase of the SDLC occurs after the product is in full operation. Maintenance of software can include software upgrades, repairs, and fixes of the software if it breaks.

**SDLC for ATM System**

* **Requirement Gathering:**

ATM System is aimed to developing an online platform on which users, access to bank account (balance, bank transfers) and retrieval of money. The requirements of the System are:

* To computerize all details regarding users account system.
* To setup a connection between the customers and the Banks.
* To give the facility to make transactions such as deposit money, withdraw, pin change and balance inquiry.
* To give the user easy transaction facility.
* **Analysis:**

In the Analysis Phase the information that is analysed is written below.

* Withdrawal: The User selects Withdraw from the menu and withdraws cash from the ATM.
* Deposit: The User selects Deposit option from the menu and deposits cash or cheques into the ATM.
* Bill Payment: The User selects Payment from the menu and enters bills into the ATM that are to be paid. The User also has the ability to enter up to 3 bills in one transaction
* Account Update: The User selects Account Update from the menu and a balance is displayed for the account.
* Print Transaction Record: ATM prints a record after a transaction.
* Exit: User completes sessions with ATM and retrieves card.
* **Design:**

This ATM system is based on the database, PHP programming language and ajax and jquery techniques. My SQL (Structure Query Language) is used in areas where keeping the records in the database is necessary and it also perform operation like insert, delete, update, edit. This system uses HTML and PHP as frontend and MySQL work as backend. In ajax technology it is asynchronous java script and xml. It is used for table loading purpose.

* **Implementation:**

This is a phase in which the system analyst did an evaluation of the changeover method that should be used to switch from present manual system to the developed computerized system. After a close analysis the analyst came up with parallel changeover method as the most appropriate for the system. Parallel method is whereby the computerized system will run concurrently with the manual system before discarding the manual system. Although expensive the changeover method will prove to be the most efficient because:

* Parallel changeover provides time for one the database administrator to update all the account information, owner’s information before a total changeover to the new system.
* It’s possible to troubleshoot any errors arising from loading process without affecting the banking transactions as the manual system will still be in place to carry out the banking activities smoothly.
* Provides time for employees to learn and adapt to the new system.
* Lowers the risk to the management in case of a technical hitch or breakdown as the manual system will still be in place as the analyst fixes the technical hitch.
* **Testing:**

One of the purposes of the testing is to validate and verify the system. Verification means checking the system to ensure that it is doing what the function is supposed to do and Validation means checking to ensure that system is doing what the user wants it to do.

* Verify that the portal for new Account Holder’s registration has all the mandatory fields required for registering a User.
* Verify that after filling the details and successful payment providing them their Debit-Card.
* Verify that after transactions based on the requirement the details are updated in account details database.
* Verify that for existing account based on account number of the user, their records are added/updated in the database.
* Verify that the system has an admin for Bank managers as well.
* Verify that new details of new managers and account holder can be added in the system.
* Verify that details of existing users can be updated in the system etc…
* **Maintenance:**

The proposed system is ATM System. We can enhance this system by including more features like providing online loans, opening online bank account etc. Providing such features enable the users to include more comments into the system.

**Conclusion:**

In above practical we study about Software Development Life Cycle and analyse various activities conducted as a part of various phases.



**SARDAR PATEL COLLEGE OF ENGINEERING-BAKROL**

Date:

**PRACTICAL: 3**

**Aim: Consider any project to be developed in any technology as a Software Architect or Project Manager. Construct Software Requirement Specification (SRS) document for the project.**

What is Software Requirement Specification [SRS]?

* A software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.
* SRS is a document created by system analyst after the requirements are collected from various stakeholders.
* SRS defines how the intended software will interact with hardware, external interfaces, speed of operation, response time of system, portability of software across various platforms, maintainability, speed of recovery after crashing, Security, Quality, Limitations etc.
* The requirements received from client are written in natural language. It is the responsibility of system analyst to document the requirements in technical language so that they can be comprehended and useful by the software development team.

SRS should come up with following features:

* User Requirements are expressed in natural language.
* Technical requirements are expressed in structured language, which is used inside the organization.
* Design description should be written in Pseudo code.
* Format of Forms and GUI screen prints.
* Conditional and mathematical notations for DFDs etc.

**Why SRS?**

* In order to fully understand one’s project, it is very important that they come up with a SRS listing out their requirements, how are they going to meet it and how will they complete the project. It helps the team to save upon their time as they are able to comprehend how are going to go about the project. Doing this also enables the team to find out about the limitations and risks early on.

## Types of Requirements:

* The below diagram depicts the various types of requirements that are captured during SRS.



**Software Requirement specification for Expense Manager**

**Purpose**

* The software is for managing expenses.
* It maintains two levels of user’s (1) Admin (2) User.
* The software includes maintaining user’s daily expenses.
* Advertising, recommending and notifications.
* Bill generation.

**Scope**

* The proposed software product is a online managing system. The system will be used to get the expense details from the user and then display that data for the future use.
* The current system in use is a hand to hand based system. It is too slow and cannot provide updated lists of expense within a reasonable timeframe.
* Requirements statements in this document are both functional and non-functional.

**Overview**

* This Software Requirements Specification (SRS) is the requirements work product that formally specifies Expense manager.
* It includes the results of both business analysis and systems analysis efforts Various techniques were used to elicit the requirements and we have identified your needs, analysed and refined them.
* The objective of this document therefore is to formally describe the system’s high level requirements including functional requirements, non-functional requirements and business rules and constraints.

**General Description**

* Product Perspective
* This expense managing system is a self-contained system that manages expenses of the user Info. Various stakeholders are involved in the users info system.
* Product features
* The system functions can be described as follows:
* Advertisement: Our application let user post the Ad of his/her product which he/she wants to put on rent.
* Recommendation: Our application recommends related expenses to the user.
* Contact and Customer Care: Our application would be providing the email id of the customer care centre and other contacting options via calls.
* Design and Implementation Constraints
* Database: The system shall use the MySQL Database, which is open source and free.
* Operating System: The Development environment shall be Windows 8+.

**Interface Requirements**

* User Interface
* The software provides good graphical interface for the user. He/she can operate the system easily by performing the required task such as creates, update, viewing the details of the expenses.
* Allows user to view quick reports like time to time expenses etc. in between particular time.
* Price verification and search facility based on different criteria.
* Hardware interface
* Operating system : windows 8+.
* Hard disk : 250 GB
* RAM : 4 GB
* Processor : AMD A10
* Software interface
* HTML, CSS, PHP, JavaScript, .Net.
* Sublime Text
* My SQL server
* Communication interface
* Application

**Non- functional Requirement**

* Performance
* Response Time: - The system shall give responses in 5 second after checking the expense details.
* Capacity: - The System supports 1 people at a time.
* User-interface: - The user-interface screen shall respond within 5 seconds.
* Conformity:- The systems must conform to the Microsoft Accessibility
* Security
* Modification:-Any modification (insert, delete, update) for the Database shall be synchronized and only by the user itself.
* Reliability
* How general the form generation language is Simplicity vs. functionality of the form language= Speeds up form development but does not limit functional.
* Availability
* The system shall be available all the time.
* Safety
* Humans are error-prone, but the negative effects of common errors should be limited. E.g. Users should realize that a given command will delete data, and be asked to confirm their intent or have the option to undo.
* Software Quality
* Good quality of the framework= produces robust, bug free software which contains all necessary requirements for user satisfaction.
* Reusability
* Is part of the code going to be used elsewhere= produces simple and independent code modules that can be reused.
* Maintainability
* Back Up The system shall provide the capability to back-up the Data.
* Errors the system shall keep a log of all the errors.

**Conclusion:**

In above practical we study about Software Requirement Specification (SRS) and also write software requirement specification for Expense manager.



**SARDAR PATEL COLLEGE OF ENGINEERING-BAKROL**

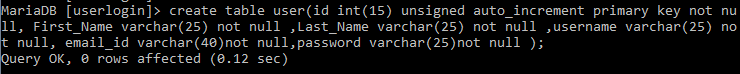
Date: 28/01/19

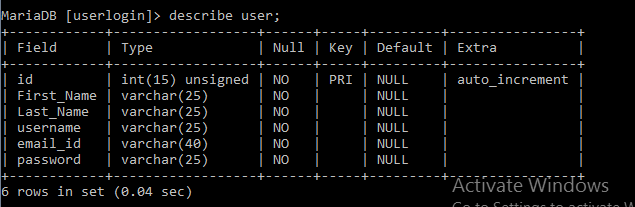
**PRACTICAL: 4**

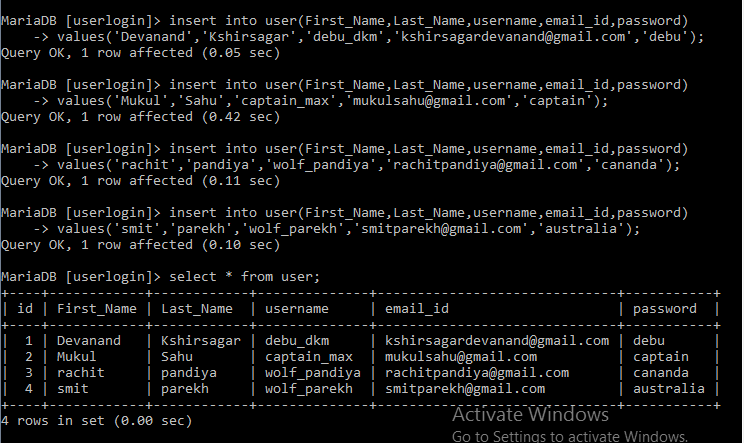
**Aim: Consider your project and make a database related to the project and use any query related to database to find the data you have entered.**

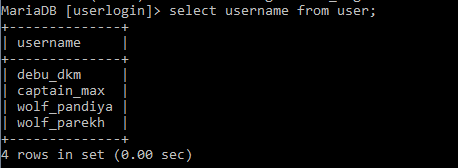
1] Create Database for User Login.

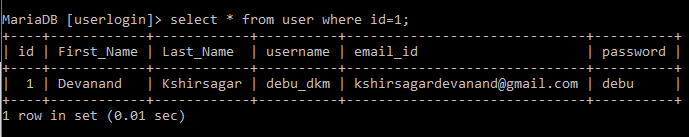


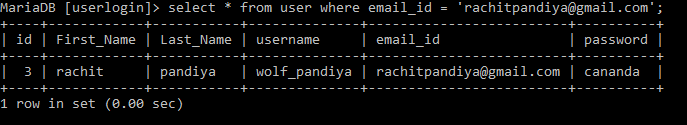
2] Create Table for user to Enter the details.



3] Insert the information of user.

4] Write a Query to find the username of the users.

5] Write a Query to get the data from the id.

6] Write a Query to get the person details by his Email\_ID.

**Conclusion :-**

In above practical we study about the database and how to make the database and we also learn many queries to make a change to database and to handle it efficiently.