**INTRODUCTION**

**QR Code**. A **QR code** (short for "quick response" **code**) is a type of **barcode** that contains a matrix of dots. It can be scanned using a **QR scanner** or a smartphone with built-in camera. Once scanned, software on the device converts the dots within the **code** into numbers or a string of characters.

(**Q**uick **R**esponse code) A two-dimensional barcode widely used for many purposes. When scanned with a mobile tagging app in a smartphone, the QR code can cause a Web page or app to download or even capture business card data. Codes are found in newspapers, magazines, brochures and store shelves, and billboard-size codes can be scanned at a distance, because the QR app uses the phone's camera.

**4K of Alphabetic Data**  
The QR code can store up to 4,296 alphanumeric or 7,089 numeric characters, and if a high level of error correction is used, up to 30% of the image can be smudged and still be recognized. Created in the mid-1990s by a Toyota subsidiary to track parts on assembly lines, they are widely used in Japan. The technology became an ISO standard in 2000 and gained popularity in the U.S. within the decade.

**Generation procedure**

The process (and high-level algorithm) for generating a QR Code symbol is as follows:

1. Choose the text (Unicode string) or binary data (byte string) to encode.
2. Choose one of the 4 error correction levels (ECL). A higher ECC level will yield a barcode that tolerates more damaged parts while preserving the payload data, but will tend to increase the version number (i.e. more modules in width and height).
3. Encode the text into a sequence of zero or more segments. A segment in byte mode can encode any data, but using alphanumeric or numeric mode is more compact if the text falls into these subsets.
4. Based on the segments to be encoded and the ECL, choose a suitable QR Code version to contain the data, preferably the smallest one.
5. Concatenate the segments (which have headers and payload) and add a terminator. The result is a sequence of bits.
6. Add padding bits and bytes to fill the remaining data space (based on the version and ECL).
7. Reinterpret the bitstream as a sequence of bytes, then divide it into blocks. Compute and append error correction bytes to each block. Interleave bytes from each block to form the final sequence of 8-bit codewords to be drawn.
8. Initialize a blank square grid based on the version number.
9. Draw the function patterns (finders, alignment, timing, version info, etc.) onto the appropriate modules. This is formatting overhead to support the QR Code standard, and does not encode any user data.
10. Draw the sequence of (data + error correction) codewords onto the QR Code symbol, starting from the bottom right. Two columns at a time are used, and the scanning process zigzags going upward and downward. Any module that was drawn for a function pattern is skipped over in this step.
11. Either manually or automatically choose a mask pattern to apply to the data modules. If masking automatically, then all 8 possibilities are tested and the one with the lowest penalty score is accepted. Note that the format information is redrawn to reflect the mask chosen.
12. We are now finished the algorithmic parts of QR Code generation. The remaining work is to render the newly generated barcode symbol as a picture on screen, or save it as an image file on disk.

**PROJECT ANALYSIS**

1. **PURPOSE OF THE PROJECT**
   * Provide a Single environment from where an administrator can interact with the system resources with in the network.
   * Provides Info regarding the no of services running on the system.
   * To provide an upgrade support for the system resources.
   * Provides interface to interact with the system parameters like network settings, OS Settings, Services, and Process etc.
   * Provides means of reporting the system setting to a person through email. There by an administrator can know the status of a systems throughout the network
2. **SOLUTION OF PROBLEM PROBLENS**

* As Snack wizard is a tool. It requires the settings to be done by the administrator, in order to allow the remote access to the resources.
* Three different Categories are provided to know the current status of the services, drivers etc.
* The single application can be used to configure multiple versions of operating systems which is possible by allowing the application settings to get configures and show appropriate panel for the administrator.
* Updating of components can be done by accessing the components of the system as well as the resources available on the game .

## PERT CHART

PERT CHART stand for Program Evaluation Review Techniques, unlike bar chart PERT can be both a cost and a time management system. PERT is organized by events and activity or task. One advantage of the PERT chart is that it is a schedule device that also shows graphically which task must be completed before others are begun.   
  
Advantage of the pert chart is as follows:

* It force the manager to the plan
* It shows the interrelationship among the tasks in the project and in particularly it identified the critical path of the project, thus helping to focus on it.
* It exposes all the particularly all-possible parallelism in the activities and thus help in allocating the resources.
* It allows scheduling and simulation of alternative schedules.
* It enables the manager to monitor and control the project.

**Design of the system**

The design of an information system produces the details that clearly describe how a system will meet the requirements identified during system analysis. Systems specialists often refer to this stage as logical design, in contrast to the process of developing program software, which is referred to as physical design.

**Devlepment of the softwere:-** Software development may install purchased software or they may develop new, custom designed programs. The choice depends on the each option, the time available to develop software and the availability of programmers. Generally it has been observed that programmers are part of permanent staff in a big organization. In smaller organization, without programmers, outside programming services may be hired or retained on a contractual basis. Programmers are also responsible for documenting the program, providing an explanation of how and why certain procedures are coded in specific ways. Documentations is essential to test the program and carry on maintenance once the application has been installed.

**SYSTEM ANALYSIS**

System analysis:-A System is the collection of interrelated components that works together to archive some common objective and system analysis is the specification as what the system required to do. It is a management technique which helps us in designing a new system or improving an existing system.

A system analyst should have various skills to effectively carry out the job specifically. These skills can be divided into two categories. These are interpersonal skills and technical skills interface of the analyst with people in business. They are useful in establishing trust, resolving conflict, and communication information. Technical skills on the other hand, focus on the procedures and techniques for operations analysis, system analyst and computer science.

The interpersonal skills which are relevant to systems work are following:-

1. Communication: - Communication is not just reports, telephonic conversations, and interview. It is people talking, listening, felling, and reacting to one another, their experience and reaction. Some indicators of one another, their experience and reaction to the, their experience and reactions. Some indicator of climate of closed communication is defensive memos, excessive correspondence, and failure to speak up for fear of being identified. Therefore, opening communication channels are a must for system development.
2. Understanding: - Identified problems and assessing their remedies is one of the attributes of good system analyst. A system analyst should have the grasp of company goals and objective.
3. Teaching: - A system analyst should educate people in the use of computer systems, selling the system to the user and giving the support when needed.
4. Selling: - a system analyst should have selling ideas and promoting innovations in problem solving using computer.

For system analysis we know that we operate in a dynamic environment where way of life is. To construct a system the following key elements must be considered:

**(1) Outputs and inputs**: A major objective of a system is to produce an output that has value to its user. Whatever the nature of the output (goods, services, or information), it must be in line with the expectations of the intended user. Inputs are the elements (material, human recourses, information) that enter the system for processing. Output is the outcome of processing. A system feeds on input to produce output in much the same way that a business brings in human, financial, and material resources to produce goods and services. Output is a first step in specifying the nature, amount, and regularity of the input needed to operate a system. Input and processing design follow.

Compare Output

Management Control

Services

Human re

Transformation

Standard of performance

Input

Processing Output

**(2)** **The processor(s):** The processor is the element of a system that involves the actual transformation of input into output. It is the operational component of a system. Processor may modify the input totally or partially, depending on the specifications of the output.

**(3) Control:** The control elements guide the system. It is the decision-making subsystem that controls the pattern of activities governing input, processing, and output. In an organization context, management of the bank as a decision-making body controls the inflow, handling, and outflow of activities that affect the welfare of the business. In a computer system, the operating system and accompanying software influence the behavior of the system.

**(4) Feedback:** Feedback may be positive or negative, routine or informational. Positive feedback reinforces the performance of the system. It is routine in nature. Negative feedback generally provides the controller with information for action. In system analysis, feedback is important in different ways. During analysis, the user may be told that the problems in a given application verify his/her initial concerns and justify the need for change. Another form of feedback comes after the system is implemented. The user informs the analyst about the performance of the new installation. This feedback often results in enhancements to meet the user’s requirements.

**(5)Environment:-**The environment is the “super system” within which library operates. It is the source of external elements that impinge on the system. In fact, it often determines how a system must function such that the Bank’s environment, consisting of customers and others.

**(6)Boundaries and Interface:** A system should be defined by its boundaries –the limits that identify its components, process, and interrelationships when it interfaces with another system.

There are generally eight types of System analysis technique:-

**1. Requirement technique**:- Requirement determination is generally done through Extensive study of the system includes the understanding of the goal. Process And constraints of the system for which information are designed. Several Forms are also designed and illustrated in the text of system analysis. There are no straight forward algorithms to elicit the requirement from the user. It is an iterative process, which the analyst use while interviewing several user groups. It is art rather then science.

**2. Diagrammatic technique**:- Data flow document flow diagrams represent perhaps most the most widely used diagramming technique of the system analyst. The document flow diagram graphically represents the various documents that flow across the system.: the information carried by the paper document must be generated and proceed by proposed information system.

**3. Data flow diagram(DFD)**:- Data flow diagram is a powerful diagram that can be used to document the information flow. It also presents itself to be broken down in top down fashion .At the top level ,data flow are represented at the very abstract aggregate level .Each component of the data flow is further broken down to different levels, so that at each level. We have just a few entities to concatenate on , DFD have developed a representation scheme to represent data store. Process(where some changes are made to system) and entities(the player in the game) and the actual information flows.

**4. Data Dictionary**:-Another powerful tool that is extensively used in system

analysis in the data dictionary. DDs provides a detailed reference to every data item—the different names by which the item is represented in different program modules, different data structure used to represent the item in different modules. The module where the date item is generated ,where it is stored destroyed .In essence it provides a quick snapshot of every data item is generated ,where it is stored and destroyed .In essence it provides a quick snapshot of every date item is used by the information system. In essence it provides a quick snapshot of every data item used by the information system .It is very useful for consistency checks. System modification and completeness checking.

While these techniques are general in nature and by the analyst in the different stages of the system life-cycle the following are specific to some of the steps of the system life-cycle.

**5.Feasibility Report**:- A typical structure of the feasibility report will be under:-

A preamble that sets the stage for the project followed by goals statement that quantity precisely the goals of the proposed information system. This is followed by a short narrative that describe in unambiguous yet jargon free Language the present system. This is understandable to any intelligent person not necessary a computer professional or a even a computer literate .the proposal alternative are then describe once again in a reasonably jargon free language . Being a feasibility study the alternative are unlikely to be detailed to the full extent. Until full system is developed in its entirety, the full detail are unlikely to be known. Yet we can’t go ahead with the final system without doing a feasibly analyses. The detail of the system to be built may bum ford gradually from understood by the user as well as analyst. Based on ‘sketchy’ design of the proposed alternatives, an order of magnitude cost benefit study is preferred. The end-user decides a particular alternative that is worked out in detail for further implementation. The detail design phase starts here.

**6. Detailed design:** - The detailed specifications are worked out followed by hardware/software plan. This constitutes system designs which once again need to be whetted by the user. Once this is done detailed system design starts. Effectively the analysis phase ends here and the design phase begins. It may involve substantial effort on the part of technical system analyst, hardware, software, communication specialists etc. a major component of the detailed system design is the database design actual coding is undertaken after the database design is completed.

1. **Database Design**: - DBMS permits efficient storage and manipulation of data files they do not cater to the structuring of the database themselves .There is the need for the right abstraction of data into the database so that any update/query operation captures the spirit of the data stored in the database. Normalization is used which leads to the decomposition in such a way that no information is lost due to processing of data. Database theory details further degrees of normalization including 4NF and 5NF. While theoretically sound, such further refinement add (?) like, if, any, to data modeling real world data. Since our text is primarily on information system and not on database theory we will not further elaborate an advanced normalization.

**8. System implementation**:- System implementation includes the detailed design of the process, their validation and thought checking. while the formal methods of proving program correctness are evolving ,they are still not useful to test out large commercial software to help information system planning .Many of the analysts use experimental version using what is known as “Parallel runs ”. Here both the current system and proposed new system are run in parallel for a specified time period and the current system is used to validate the purposed system.

**Feasibility study:🡪**

Feasibility is the determination of the whether or not a project is worth doing. The process followed in making this determination is called a feasibility study. These types of study determine if a project can and should be taken. Once it has been determination that a project a feasible, the analyst can go ahead and prepare the project specification which finalizes project requirements. Generally, feasibility studies are undertaken within tight time constraints and normally culminate in a written and oral feasibility report . The contents and processed, postpone or cancel the project. Thus, since the Feasibility study may lead to the commitment of large resource, it becomes Necessary that it should be conducted competently and that no fundamental errors of judgment made.

Types of feasibility:🡪

1. Technical feasibility
2. Operational feasibility
3. Economical feasibility
4. Social feasibility
5. Management feasibility
6. Legal feasibility
7. Time feasibility
8. **Technical feasibility**:- Technical feasibility involves determining whether or not a system can actually be constructed to solve the problem at end. This is considered with specifying equipment and software that will successfully satisfied the user requirement. The technical needs of the system may vary considerable ,but might include:-

🡪The facility to produce outputs in a given time.

🡪Response time under certain condition.

🡪Ability to process a certain volume of transaction at a particular Speed.

🡪Facility to communicate data to distant location.

1. **Operational feasibility**: - Proposed projects are of course beneficial only if they can be turned into information system that will meet the organization’s operating requirement. Simply started, these tests of feasibility ask the system will work when developed and installed. There are major barrier for implementation. It is related to human organizational and political aspects. The points to be considered are :-

🡪 What change will be brought with the system?

🡪 What organizational structures are distributed?

🡪 What new skills will be required? Does the existing staff member have?

These skills? If not, can they be trained in the due course of time?

1. **Economical feasibility**: - Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as cost/benefit analysis the procedure is to determine the benefits and saving that are expected from a proposed system and compare them with cost. If benefit out weights costs, a decision is taken to design and implementation the system.
2. **Social Feasibility**: - Social Feasibility is a determination of whether a proposed project will be acceptable to the people or not. This determination Typical examines the probability of the project being accepted by group

Directly affected by the proposed system change.

1. **Management feasibility**:- It is a determination of whether a proposed project will be acceptable to management. If management does not accept a project or gives a negligible support to it, the analyst will tend to view the project as a non-feasible one.

**6. Legal feasibility**: - Legal feasibility is a determination of whether a proposed project infringes on known acts, Statutes, as well as any pending legislation. Although in some instance the project might appear sound, on closer investment it may be found to infringe on several legal areas.

**7. Time feasibility**: - Time feasibility is a determination of whether a project can be implementing fully with in a stipulated time frame .If a project takes too much time it is likely to be rejected.

**COST/BENEFIT ANALYSIS**:

In developing cost estimates for a system, we need to consider several cost elements.

1. **Hardware costs** relate to the actual purchase or lease of the computer and peripherals. Determining the actual cost of hardware is generally more difficult when various users than for a dedicated stand-alone system share the system. In some cases, the best way to control for this cost is to treat it as an operating cost.
2. **Personnel costs** include EDP staff salaries and benefits as well as pay for those involved in developing the system. Once the system is installed the costs of operating and maintaining the system become recurring costs.
3. **Facility costs** are expenses incurred in the preparation of the physical site where the application or the computer will be in operation.
4. **Operating costs** include all costs associated with the day-to-day operation of the system; the amount depends on the working hour, the nature of the application.
5. **Supply costs** are variable costs that increase with increased use of paper, ribbon, disks, and the like.

**Procedure for Cost/Benefit Analysis:**

1. Identify the costs and benefits pertaining to a given project.
2. Categorize the various costs and benefits for analysis.
3. Select a method of evaluation.
4. Interpret the results of the analysis.
5. Take action.

# DATA FLOW DIAGRAMS

Data flow Diagram is concerned with designing a sequence of functional transformation that converts system inputs into the required outputs. The design is represented as data flow diagram. These diagrams illustrate how data flow through a sequence of functional transformations.

Data flow diagram is a useful and intuitive way of describing a system. They are normally understandable without special training. Especially if control information is exclude. They show end-to-end process from when data entry the system to where it leaves the system can be traced.

Notation Used in Data-Flow Diagram:-

**Rectangles**: represents external entities that interaction with the system. They provide input or receive output.

**Arrows:** Shows direction of data flow. Their name describe the data flowing

along the path.

**One side open box** : represents data stores.Again ,theyshould be given a descriptive name.

**Circles**: represents functions ,which transforms input to output, the transformation name indicates its function.

**DATA FLOW DIAGRAM**:

A data flow diagram is graphical tool used to describe and analyze movement of data through a system. These are the central tool and the basis from which the other components are developed. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow diagrams. The physical data flow diagrams show the actual implements and movement of data between people, departments and workstations. A full description of a system actually consists of a set of data flow diagrams. Using two familiar notations Yourdon, Gane and Sarson notation develops the data flow diagrams. Each component in a DFD is labeled with a descriptive name. Process is further identified with a number that will be used for identification purpose. The development of DFD’s is done in several levels. Each process in lower level diagrams can be broken down into a more detailed DFD in the next level. The lop-level diagram is often called context diagram. It consists a single process bit, which plays vital role in studying the current system. The process in the context level diagram is exploded into other process at the first level DFD.

The idea behind the explosion of a process into more process is that understanding at one level of detail is exploded into greater detail at the next level. This is done until further explosion is necessary and an adequate amount of detail is described for analyst to understand the process.

Larry Constantine first developed the DFD as a way of expressing system requirements in a graphical from, this lead to the modular design.

A DFD is also known as a “bubble Chart” has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is the starting point of the design to the lowest level of detail. A DFD consists of a series of bubbles joined by data flows in the system.

**DFD SYMBOLS:**

In the DFD, there are four symbols

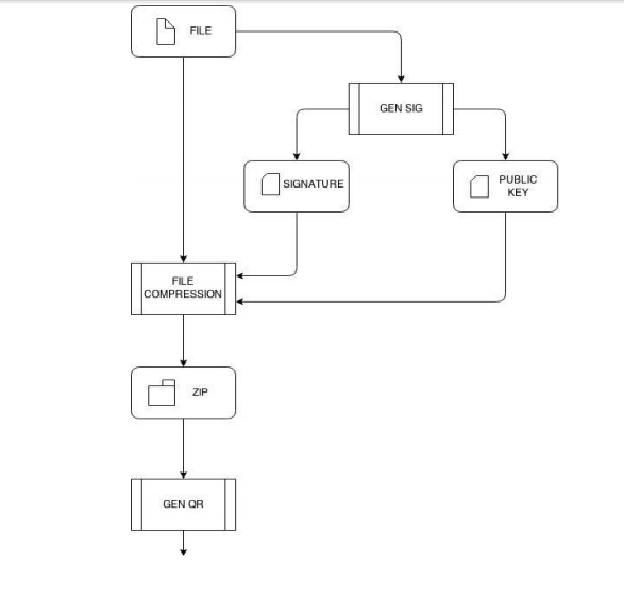
1. A square defines a source(originator) or destination of system data
2. An arrow identifies data flow. It is the pipeline through which the information flows
3. A circle or a bubble represents a process that transforms incoming data flow into outgoing data flows.
4. An open rectangle is a data store, data at rest or a temporary repository of data

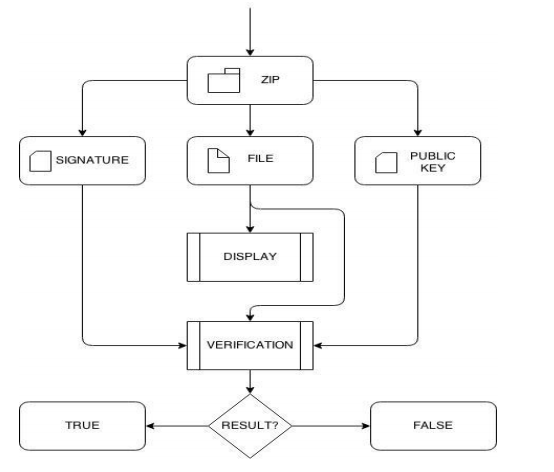
Process that transforms data flow.

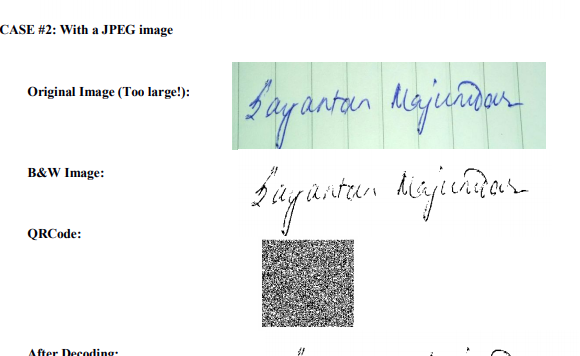
Source or Destination of data

Data flow

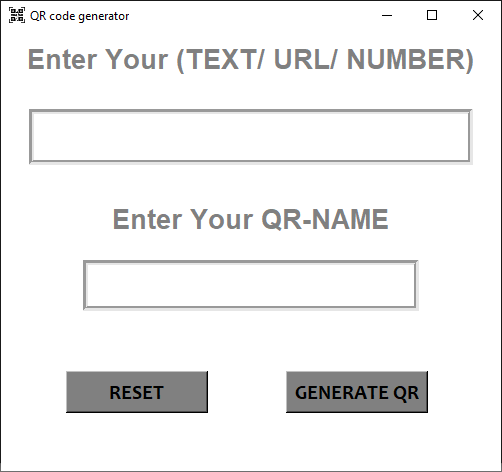
Data Store







**Coding and Screen shot**



import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()

import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

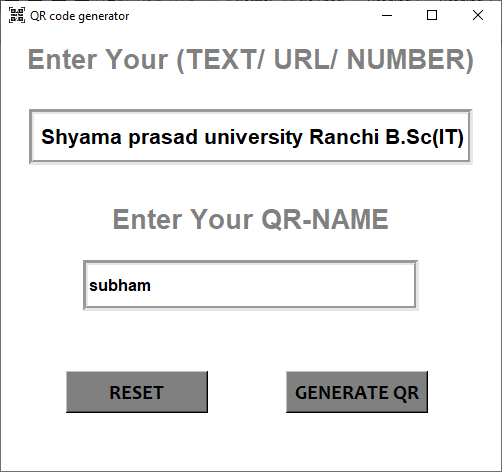
webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()



import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()

import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

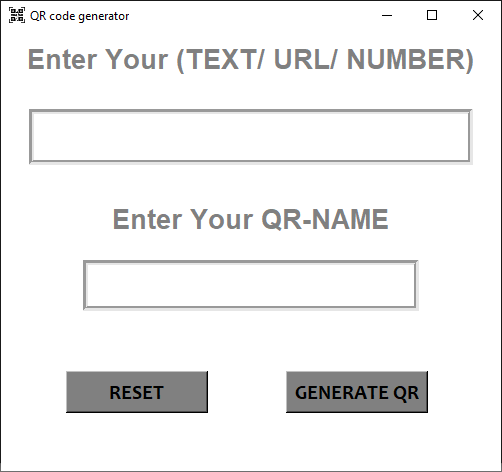
webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()

**Coding and Screen shot**



import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()

import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

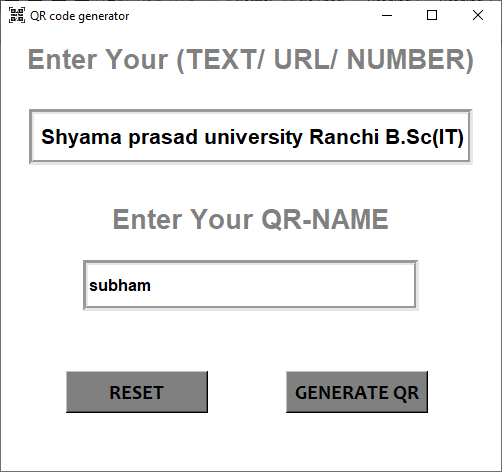
webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()



import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()

import qrcode as qq

from tkinter import \*

# from tkinter import messagebox

def reset():

    webEntry.delete(0,END)

    webEntry.config(bg='white')

    webEntry2.delete(0,END)

    webEntry2.config(bg="white")

def QRCode():

    website = webEntry.get()

    website2 = webEntry2.get()

    if len(website)<1:

        messagebox.showinfo('QR Generator','Please Enter text /url /number :)')

    if len(website2)<1:

        messagebox.showinfo('QR Generator','Please Enter file name :)')

    else:

        qr=qq.make(website)

        qr.save(website2+'.png')

        messagebox.showinfo('QR Generator','QR Code is generated Sucessfully :)')

root=Tk()

root.geometry("500x440+420+100")

root.config(bg="white")

root.iconbitmap('D:\\Python\\qrcodegenerator\\qr-code.ico')

root.title('QR code generator')

Label(root,text="Enter Your (TEXT/ URL/ NUMBER)",font="arial 20 bold",bg="white",fg="gray").pack(pady=(10,0))

webEntry = Entry(root,fg='black',bg="white",font="arial 16 bold",width=36,relief=GROOVE,bd=5)

webEntry.pack(pady=(30,30),ipady=10)

Label(root,text="           Enter Your QR-NAME",font="arial 20 bold",bg="white",fg="gray").place(x=20,y=170)

webEntry2 = Entry(root,fg='black',bg="white",font="arial 12 bold",width=36,relief=GROOVE,bd=5)

webEntry2.pack(side=BOTTOM,ipady=10,pady=(0,160))

Button(root,text="RESET",font="candara 15 bold",bg="gray",fg="black",width="12",command=reset).place(x=65,y=340)

Button(root,text="GENERATE QR",font="candara 15 bold",bg="gray",fg="black",width="12",command=QRCode).place(x=285,y=340)

root.mainloop()

**TESTING**

In software development process, errors can be injected at any stages during development. No method is perfect for software testing and it is expected that some errors of the earlier phases will finally reaches to the coding phase. This is because most of the verification methods of earlier phases are manual. Hence the code developed during coding activity is likely to have some requirement errors and design errors, in addition to errors introduced during the coding activity.

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 evaluated to determine if the program is performing as expected. From this it is clear that testing is used to find out errors rather than to tell the exact nature of the error. Also, the success of the testing process clearly depends upon the test cases used.

Testing is a complex process. In order to make the process simpler, the testing activities are broken into smaller activities. Due to this, for a project, incremental testing is generally performed. In incremental testing process, the system is broken into set of subsystems and these subsystems are tested separately before integrating them to form the system for system testing.

**WHAT IS TESTING ?**

By the definition of IEEE (Institute of Electrical and Electronics Engineers), testing is the process of analyzing a software item to detect the difference between existing and required conditions and to evaluate the features of the software items.

By the definition of Myers, testing is the process of analyzing a program with the intent of finding errors.

**WHAT ARE TESTING FUNDAMENTALS?**

*There are three main testing fundamentals. That are as follows: -*

* **Error:** The term error is used to refer to the discrepancy between computed, observed or measured value and the true or specified value. In other words, it is the difference between the actual output of software and the correct output.
* **Fault:** Fault is a condition that causes a system to fail in performing its required function.
* **Failure:** Failure is the inability of a system to perform a required function according to its specifications. A software failure occurs if the behavior of the software is different from the specified behavior.

**SOFTWARE TESTING STRATEGIES**

Testing is a set of activities that can be planned in advanced and conducted systematically. A strategy for software testing must accommodation low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

There are three types of testing strategies

1. Unit test
2. Integration test
3. Performance test

**UNIT TESTING**

Unit testing focuses verification efforts on the smallest unit of software design module. The unit test is always white box oriented. The tests that occur as part of unit testing are testing the module interface, examining the local data structures, testing the boundary conditions, execution all the independent paths and testing error-handling paths.

**INTEGRATION TESTING**

Integration testing is a systematic technique or construction the program structure while at the same time conducting tests to uncover errors associated with interfacing. Scope of testing summarizes the specific functional, performance, and internal design characteristics that are to be tested. It employs top-down testing and bottom-up testing methods for this case.

**PERFORMANCE TESTING**

Timing for both read and update transactions should be gathered to determine whether system functions are being performed in an acceptable timeframe.

**MAINTAINANCE**

Software will undoubtedly undergo change after it is delivered. Change will occur because errors have been encountered, because the software must be adapted to accommodate changes in its external environment (e.g., a change required because new operating system or peripheral device). Software can also undergo changes if the customer wants to add features based on functional or performance enhancement. The software maintenance phase may reapply each of the preceding software engineering life cycle steps to existing program rather than a new one.

Through maintenance is not considered a part of software development, it is a very important activity in the life of a software product. Usually, the total cost of maintenance phase is much higher than the development cost of the software. Normally the maintenance-to-develop cost ratio is suggested as 80:20,70:30 or 60:40.

Maintenance work is based on existing software, as compared to development work that creates new software. In other words, maintenance revolves around understanding the existing software and maintainers spend most of their time trying to understand the software they have to modify. Understanding the software means that understanding not only the code but also the related documents. During the modification of the software, the effect the change has to be clearly understood by the maintainer. To test whether those aspect of the system that are not supposed to be modified are operating as they were before modification, regression testing is done. In regression testing, we use old test cause to test whether new errors have been introduced or not.

Thus, maintenance involves understanding the existing software, understanding the effect of changes, making the changes to both code and documents, testing the new parts and re testing the old parts that were not changed. In order to make the maintainer job easier, it is necessary to prepare some supporting documents during software development, makes maintenance the most costly activity in the life of software product.

Testing is always effective with a test plan. A test plan is a general document for the entire project that defines the scope, approach to be taken& the schedule of testing as well as identifies the test items for the entire testing process & the personnel responsible for the difficult activities of testing. Inputs for forming test plan are-

1. Project plan
2. Requirement document, &
3. System design document

Project plan is needed to make sure that the test plan is consistent with the overall plan for project. Requirement document & the system design document are the document used for selecting the test units & deciding the approaches to be used during testing.

**A TEST PLAN CONTAINS THE FOLLOWING**

* **TEST UNIT SPECIFICATION** : A test unit is a set of one or more modules, together with associated data, that are from single software & that are objects of testing. A test unit may be a module or a few modules or the entire software.
* **FEATURES TO BE TESTED :**  It includes all the software characteristics specified or implied by the requirement or design documents. These may include functionality, performance, design constraints and attributes.
* **APPROACH FOR TESTING :**  It specifies the overall approach to be followed in current project. The techniques that will be used to judge the testing effort should also be specified. This is sometimes called testing criterion i.e., criterion for evaluating the set of test cases used in testing.
* **TEST DELIVERABLES :** It should be specified in the test plan before the actual testing begins. Deliverables could be a set of test cases that were used, detailed result of testing, test summary report, test log and data about the code coverage. Test summary report summarizes the results of testing activities.
* **SCHEDULE :** It specifies the amount of time and effort to be spent on different activities of testing.
* **PERSONAL ALLOCATION** : It specifies the persons responsible for performing different activities.

**PERT CHART**

**PERT CHART** stand for Program Evaluation Review Techniques, unlike bar chart PERT can be both a cost and a time management system. PERT is organized by events and activity or task. One advantage of the PERT chart is that it is a schedule device that also shows graphically which task must be completed before others are begun.   
  
**Advantage of the pert chart is as follows:**

* It force the manager to the plan
* It shows the interrelationship among the tasks in the project and in particularly it identified the critical path of the project, thus helping to focus on it.
* It exposes all the particularly all-possible parallelism in the activities and thus help in allocating the resources.
* It allows scheduling and simulation of alternative schedules.
* It enables the manager to monitor and control the project.

**Program Evaluation Review Techniques (P.E.R.T) Chart**

Design

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

Project Initiation

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

Preliminary

Investigation

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

Problem

Analysis

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

R.A

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

Decision

Analysis

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

Computation

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

Implementation

Preliminary

Investigation

s.start s.finish

Actual actual

Start Finish

**Conclusion & Future Scope:**

In cryptography the process of securing data has been improved drastically in last few years. In the proposed method the authors have used 3072-bit RSA encryption to digitally sign the file. RSA is based on the practical difficulty of factoring the product of two large prime numbers. Till date most common attack against RSA involves supercomputers with extremely large RAM space. But it has a very high initial cost and also not feasible. Of course there are other factoring methods that don't require huge supercomputers. But they are enough less efficient even at the best case situation. For example 512-bit RSA encryption created in 1999 was possible to break in 2007 with continuous effort by the scientists using modern computers. In FIPS 186-3, NIST assured that RSA with the key length of 3072-bit will remain secured beyond 2030. Though RSA is an extremely secure cryptosystem till date and it will remain secure for a long while, the scenario may change quickly with the fast growth of technology and science. It is clear that, to maintain the security of this scheme we must update the crypto-system regularly. From the practical aspect, this may be a challenge to this project because of the compatibility issues. The present scheme has limitation over the storage capacity of the QR Code™. In terms of capacity QR Code™ is better than UPC (Universal Product Code) barcodes. Still, even with low error detection level it's not enough to store file greater than ~2.9KB. Though it is not impossible to split a large file and generate more than one QR Code™, it is neither always feasible nor recommended. The smartphone has a limited resource and processing power which is another bottleneck of any kind of massive computation. Scanning large QR Code™ with low error detection level in different lighting environments require a good built-in camera, which may not be available with every Android devices. The authors are already trying to incorporate some efficient cryptographic method but with less computation time than RSA.

**References:**

[1]. Zxing Library: <https://github.com/zxing/zxing/>

[2]. Java security libraries: <http://docs.oracle.com/javase/7/docs/api/java/security/package-summary.html>

[3]. Android Libraries: <https://developer.android.com/reference/packages.html>

[4]. FIPS 186-3: <http://csrc.nist.gov/publications/fips/fips186-3/>

[5]. NIST Standard: <http://csrc.nist.gov/publications/nistpubs/800-57/>

[6]. RSA reference: http://people.csail.mit.edu/rivest/Rsapaper.pdf