Palestine Polytechnic University

College of Engineering and Technology
Electrical and Computer Engineering Department

Graduation Project

Computer security algorithms simulator

Project Team

Bayan A.Ihshish

Project Supervisor

Dr. Radwan Tahboub

Hebron – Palestine
Dec, 2011

Palestine Polytechnic University

College of Engineering and Technology
Electrical and Computer Engineering Department

Graduation Project

Computer security algorithms simulator

Project Team

Bayan A.Ihshish

Project Supervisor

Dr. Radwan Tahboub

Hebron – Palestine
Dec, 2011

Hebron – Palestine
Dec, 2011
جامعة بوليتكنك فلسطين

الخليل – فلسطين

كلية الهندسة و التكنولوجيا

دائرة الهندسة الكهربائية والحاسوب

اسم المشروع

Computer Security Algorithms Simulator

اسم الطالبة

بيان احشيش

بناء على نظام كلية الهندسة والتكنولوجيا و إشراف المشرف المباشر على المشروع وموافقة أعضاء اللجنة الممتحنة تم تقديم هذا المشروع إلى دائرة الهندسة الكهربائية و الحاسوب و ذلك للوفاء بمتطلبات درجة البكالوريوس في الهندسة تخصص هندسة أنظمة الحاسوب.

توقيع المشرف

.....

توقيع اللجنة الممتحنة

....

توقيع رئيس الدائرة

.....

Dedication

To my parents

To my sisters and my brothers

To my husband

To all my families and to all my friends

To Palestine the sanctify land

Abstract

An important element in many computer security services and applications is the use of cryptographic algorithms and tools, this project provides interactive simulator learning environment for cryptography tools and algorithms.

This project gives an overview of cryptographic tools and algorithms, then for each type of tools, introduces, simulates example of the most important standardized algorithms in common use, as data encryption standard (DES) as example of symmetric encryption algorithms, RSA as example of asymmetric encryption algorithms, SHA_1 as example of hash function.

Hebron – Palestine
Dec, 2011
جامعة بوليتكنك فلسطين

الخليل – فلسطين

كلية الهندسة و التكنولوجيا

دائرة الهندسة الكهربائية والحاسوب

اسم المشروع

Computer Security Algorithms Simulator

اسم الطالبة

بيان احشيش

بناء على نظام كلية الهندسة والتكنولوجيا و إشراف المشرف المباشر على المشروع وموافقة أعضاء اللجنة الممتحنة تم تقديم هذا المشروع إلى دائرة الهندسة الكهربائية و الحاسوب و ذلك للوفاء بمتطلبات درجة البكالوريوس في الهندسة تخصص هندسة أنظمة الحاسوب.

توقيع المشرف

.....

توقيع اللجنة الممتحنة

....

توقيع رئيس الدائرة

.....

Dedication

To my parents

To my sisters and my brothers

To my husband

To all my families and to all my friends

To Palestine the sanctify land

Abstract

An important element in many computer security services and applications is the use of cryptographic algorithms and tools, this project provides interactive simulator learning environment for cryptography tools and algorithms.

This project gives an overview of cryptographic tools and algorithms, then for each type of tools, introduces, simulates example of the most important standardized algorithms in common use, as data encryption standard (DES) as example of symmetric encryption algorithms, RSA as example of asymmetric encryption algorithms, SHA_1 as example of hash function.

Abstract (Arabic)

تعتبر خوارزميات التشفير إحدى أهم العناصر في امن الحاسوب والمعلومات، وفي الوقت الذي تتزايد فيه كمية المعلومات المتدفقة عبر الشبكة العنكبوتية (الانترنت)، أصبحت الحاجة ماسة إلى فهم ومعرفة كيفية استخدام خوارزميات التشفير، في هذا المشروع تم بناء بيئة تعليمية تحاكي أهم واشهر تلك الخوارزميات، وتستخدم برامج الوسائط المتعددة لإيضاح عملها.

هذا المشروع يعطي بداية نبذة عن أهم أنواع هذه الخوارزميات، ولان عددها كبير تم اختيار خوارزمية من كل نوع، لمعرفة الية عمل كل نوع، وكيفية استخدامه، وأهمية تطبيقه في مجالات الحياة المختلفة.

Table of Contents

List of Table

List of Figure

1

Chapter one

Introduction

1.1 Overview

Interest in education in computer security and related topics has been growing at a dramatic rate in recent years. This interest has been encouraged by a number of factors, two of which stand out:[3]

1. As information systems, databases, and internet_based distributed systems and communication have become pervasive in the world, coupled with the increased intensity and sophistication of security_related attacks, organizations now recognize the need for a comprehensive security strategy. This strategy encompasses the use of specialized hardware and software and trained personal to meet that need.
2. Computer security education, often termed information security education or information assurance education has emerged as a national goal in many countries with national defense and homeland security implication.

Accordingly, this project gives an overview for one of the most important element in many computer security services and applications which is cryptographic algorithms and tools, and provides interactive simulator learning environment for cryptographic tools and algorithms.

1.2 Problem Statement

Security is becoming an everyday concern for a wide range of electronic systems that manipulate, communicate, and store sensitive and private data, also to face the increased sophistication of security related attacks, a lot of technical areas of computer security have been developing, many of this technical measures rely heavily on encryption and other types of cryptographic algorithms and tools.

This project applies bloom's taxonomy of learning domains to provide interactive simulator learning environment for cryptographic tools and algorithms, For each type of tools and algorithms, introduces and simulates example of the most important standardized algorithms in common use, as a way of learning for student and interested people.

This project using program languages and multimedia program Silverlight to show and understand how this algorithms work, in addition to interact with algorithms to find new or similar algorithms.

1.3 Project Objectives

The project has the following objectives:

1. Identifying some of the security tools and algorithms.
2. Using simulation to show and understand how the security algorithms work.
3. Designing interactive environment for cryptographic tools and algorithms, to use with face to face learning or self reading material.
4. Making the tool inexpensive or freely available.

1.4 Literature Review

There is an overwhelming amount of material, including books, papers, and online resources, on computer security, in this section mentioned the most useful reading and web sites that having and analyzing topics that related to cryptographic algorithms and tools, as:

1. Cryptography and network security: principles and practice, William Stalling, which coverage cryptographic algorithms in greater detail.
2. Applied cryptography, Schneier, is valuable reference work, it contains description of virtually every cryptographic algorithms and protocol in use up to the time of the book's publication.
3. Privacy and authentication: an introduction to cryptography, by Diffie, w, and Hellman, M.

An examples of web sites:

1. www.cryptographycode.com , provides useful collection of software.
2. www.cryptocorner.com , Simon singh's web site, contains Lots of good information, plus interactive tools for learning about cryptography.

The importance of this project that it doesn't confine on the transfer of the principles and techniques, but in applying bloom's taxonomy with all it's educational domains (knowledge, attitude, skills) to provide interactive simulator learning environment for cryptographic tools and algorithms .

1.5 Time Plan

The time plan views the stages of designing and implementing the project. Table 1.1 and Table 1.2 show the tasks scheduled for the first and second semesters.

First semester

Table 1.1 Schedule Table.

Tasks \ Weeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Project Determination																
Data Collection																
Studying Requirements																
Analysis and design																
Documentation																

Second semester

Table 1.2 Schedule Table.

Tasks \ Weeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Project Implementation																
Project Deployment																
Project Test																
Documentation																

1.6 Estimated Cost

This section lists the overall cost of the project, the cost includes the software cost, and the human recourses budget.

Software costs:

includes the costs of software used to implement the project. Table 1.3 shows there costs.

Table 1.3 Software Cost.

1	Microsoft Visual Studio.NET 2010	$2
2	Websites	$100
	Total cost	$102

1.7 Risk Management

The implementation of any project may face some risks during each stage of the project . This section illustrates what are the risks expected to face the project and what are the solutions for these risks.

1.7.1 Expected Risks

Software Risks:

- Undesirable and unexpected software errors.

Human Risks:

- An illness during the stages of the project.
- Lack of training on tools.

1.7.2 Risk Avoidance

The following strategies will be taken to avoid risks mentioned in the previous section :

- Ordering the tools in early time, and training on it.
- Good estimation of the project requirements and costs.

1.8 Report Roadmap

This report consists of six chapters, the following is a brief description of the topics that are covered in each chapter.

Chapter One : Introduction

Demonstrates an overview about the project, project requirements, objectives, time plan, estimated costs and risk management.

Chapter Two: Theoretical Background

Focuses on theories and materials that are related to the project.

Chapter Three : Design Concepts

This chapter describes project objectives in details, introduces the design options, shows the general block diagram.

Chapter four : Software System Design

This chapter presents project software in more details, it presents design options and tools that used in this project, and the classes that founded in this project.

Chapter five: Implementation and Testing

This chapter indicates how implement and testing parts of system to achieve the goals of the project.

Chapter six : Conclusions and Recommendations

This chapter describes the conclusions that conclude from this project, and recommendations and future work.

2

Chapter Two

Theoretical background

2.1 Introduction

2.2 Bloom's Taxonomy

2.3 Cryptographic Algorithms And Tools

2.4 Using Multimedia In education

2.1 Introduction

In this chapter we will focus on theories and materials that are related to the project, since this project acts a learning environment for cryptographic tools and algorithms, the second section will introduce the bloom's taxonomy in learning domains that will be followed in this project, and it's probably considered one of the most widely applied in use today, so in this section we will define the domains of educational activities that identified by Bloom, and the major categories in each domain, that this project aims to attains them in it's learning environment.

The third section in this chapter will talk about cryptographic algorithms and tools, which acts the subject of this project, so this section will introduce the basic types of cryptography algorithms, and for each type of algorithms will define examples of the most important standardized algorithms in common use, that will be simulated next.

Since this project provides interactive simulator learning environment, it prefers to use multimedia as common tools for interactive learning, so the final section of this chapter talks about using multimedia in education, and the types of media into the learning environment, also the benefits of using multimedia in education.

2.2 Bloom's taxonomy of learning:

In 1956, Benjamin Bloom headed a committee of educational psychologist who developed a classification of levels of intellectual behavior important in learning, this classification identified three domains of educational activities:[1]

1. Cognitive: mental skills (Knowledge)
2. Affective: growth in feeling or emotional areas (Attitude)
3. Psychomotor: manual or physical skills (Skills)

Since the work was produced by higher educational the words tend to be a little bigger than we normally use, so trainers often refer to these three categories as KSA

(Knowledge, Skills, and Attitude), this taxonomy of learning can be thought of as "The goals of the learning process".

The committee also produced an elaborate compilation for the cognitive and affective domains, but none for the psychomotor domain, their explanation for this over sight was that they have little experience in teaching manual skills.

Bloom's taxonomy divides the cognitive and affective domain into subdivisions starting from the simplest behavior to the most complex, the divisions outlined are not absolute and there are other systems and hierarchies that have been devised in the educational and training world, however, Bloom's taxonomy is easily understood and is probably the most widely applied one in use today.

2.2.1 cognitive domain

The cognitive domain involves knowledge and the development of intellectual skills. Bloom identified six levels of the cognitive domain, starting from the simplest behavior to the most complex, the categories can be thought of as degrees of difficulties, that is the first one must be mastered before the next one can take place.[2]

The major categories are:

1. **Knowledge**: Recall data or information.
2. **Comprehension (understanding)**: Understand the meaning, translation, interpolation, and interpretation of instructions and problems.
3. **Application**: Use a concept in a new situation or unprompted use of an abstraction.
4. **Analysis**: Separates material or concepts into component parts so that its organizational structure may be understood.
5. **Synthesis**: Builds a structure or pattern from diverse elements. Put parts together to form a whole, with emphasis on creating a new meaning or structure.
6. **Evaluation**: Make judgments about the value of ideas or materials.

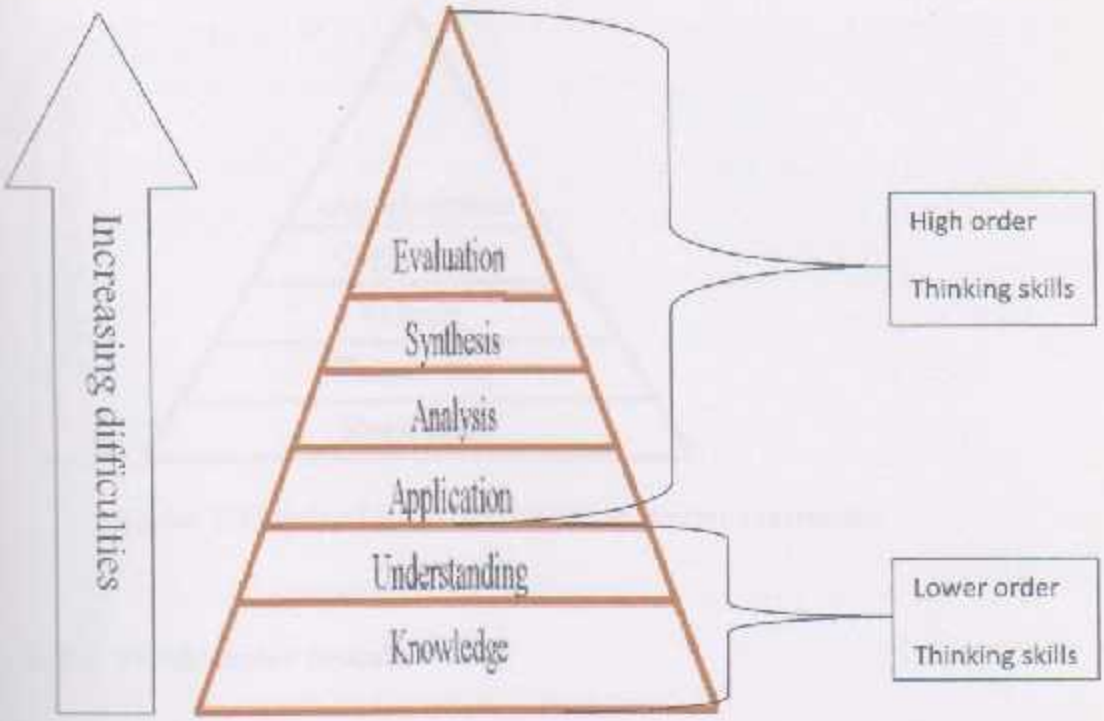

Figure 2.1 levels of cognitive domain in Bloom's taxonomy

2.2.2 Affective domain

The affective domain includes the manner in which we deal with things emotionally, such as feeling, values, appreciation, motivation, and attitude, five major categories are defined in affective domain, they are listed below from the simplest behavior to the most complex.[2]

1. **Receiving Phenomena**: Awareness, willingness to hear, selected attention.
2. **Responding to Phenomena**: Active participation on the part of the learners, attends and reacts to a particular phenomenon.
3. **Valuing**: The worth or value a person attaches to a particular object, phenomenon, or behavior.
4. **Organization**: Organizes values into priorities by contrasting different values, resolving conflicts between them, and creating an unique value system.
5. **Internalizing values** (characterization): Has a value system that controls their behavior.

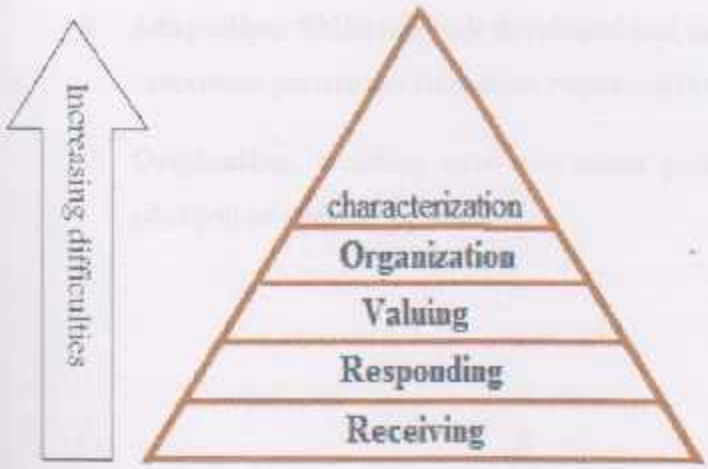

Figure 2.2 levels of Affective domain in Bloom's taxonomy

2.2.3 Psychomotor domain

As mentioned earlier, the committee did not produce a compilation for the psychomotor domain model, but other have as Simpson's version, that we talk about.

The psychomotor domain includes physical movement, coordination, and use of the motor skill areas, development of these skills requires practice and is measured in terms of speed, precision, distance, procedures, or techniques in execution, the seven major categories are listed from the simplest behavior to the most complex.[2]

1. **Perception**: The ability to use sensory cues to guide motor activity.
2. **Set**: Readiness to act. It includes mental, physical, and emotional sets.
3. **Guided Response**: The early stages in learning a complex skill that includes imitation and trial and error.
4. **Mechanism**: This is the intermediate stage in learning a complex skill. .
5. **Complex Overt Response**: The skillful performance of motor acts that involve complex movement patterns.

6. **Adaptation**: Skills are well developed and the individual can modify movement patterns to fit special requirements.
7. **Origination**: Creating new movement patterns to fit a particular situation or specific problem.

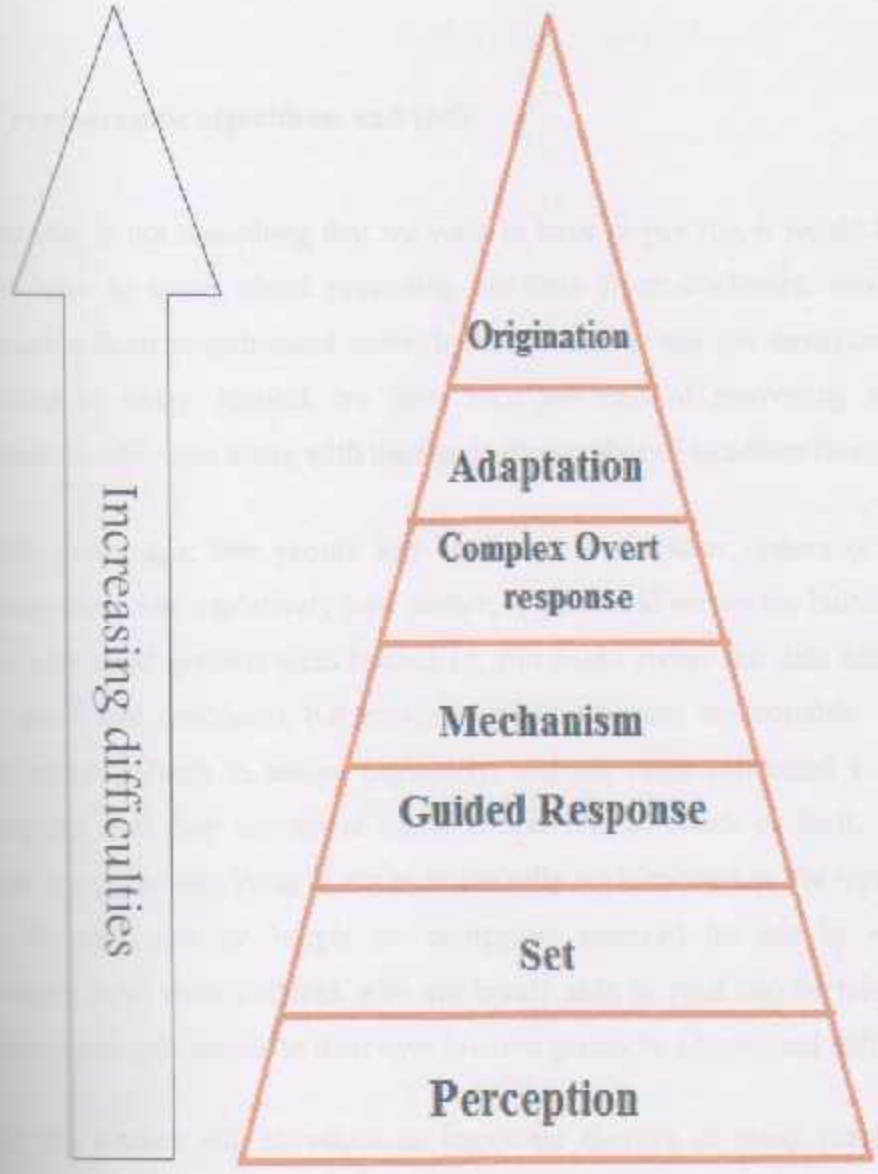

Figure 2.3 levels of psychomotor domain in Simpson version

As mentioned, the levels at the bottom are considered to be the easiest levels, so in many fields of education there is focusing on the bottoms levels of education, that are recall of information, comprehending information and applying information.

But this project will use bloom's taxonomy with all it's educational domains (knowledge, attitude, skills), so planning and designing of this project aim to teach bloom's taxonomy across most of it's levels, as will illustrate in details in chapter three when talking about intended learning outcome.

2.3 Cryptographic algorithms and tools

Security is not something that we want to have to pay for; it would be nice if we didn't have to worry about protecting our data from disclosure, modification, or destruction from unauthorized individuals, but that is not the environment we find ourselves in today. Instead, we have seen the cost of recovering from security incidents steadily rise along with the rise in the number of incidents themselves.

Fifty years ago, few people had access to a computer system or network, so securing them was a relatively easy matter, if you could secure the building that these early, very large systems were housed in, you could secure the data and information they stored and processed, but now, personal computers are portable, making them much more difficult to secure physically, and are often connected to the Internet, putting the data they contain at much greater risk of attack or theft, similarly, the typical computer user today is not as technically sophisticated as the typical computer user 50 years ago, no longer are computers reserved for use by scientists and engineers; now, even children who are barely able to read can be taught to boot a computer and gain access to their own favorite games or educational software.

In this section will introduce an important element in many computer security services and applications, which is cryptographic tools, but before it we will provide an overview of computer security and the objectives of it.

2.3.1 Computer Security

> *"The only real security that a man can have in this world is a reserve of knowledge, experience and ability".*
> —HENRY FORD

Computer security is an ever-changing issue, fifty years ago, computer security was mainly concerned with the physical devices that made up the computer, at the time, these were the high-value items that organizations could not afford to lose.

Today, computer equipment is inexpensive compared to the value of the data processed by the computer, now the high-value item is not the machine, but the information that it stores and processes, this has fundamentally changed the focus of computer security from what it was in the early years, today the data stored and processed by computers is almost always more valuable than the hardware.

Defining "computer security" is not trivial, the difficulty lies in developing a definition that is broad enough to be valid regardless of the system being described, yet specific enough to describe what security really is, in a generic sense, security is "freedom from risk or danger", so there is many meanings and related terms, one of them:

Computer Security: The protection afforded to an automated information system in order to attain the applicable objectives of preserving the **integrity**, **availability** and **confidentiality** of information system resources (includes hardware, software, firmware, information/data, and telecommunications).[3]

The "CIA" of Security:

Almost from its inception, the goal of computer security has been threefold: confidentiality, integrity, and availability—the "CIA" of security.

The purpose of confidentiality is to ensure that only those individuals who have the authority to view a piece of information may do so. No unauthorized individual should ever be able to view data they are not entitled to access.

Integrity is a related concept but deals with the generation and modification of data. Only authorized individuals should ever be able to create or change (or delete) information.

The goal of availability is to ensure that the data, or the system itself, is available for use when the authorized user wants it.

As a result of the increased use of networks for commerce, two additional security goals have been added to the original three in the CIA of security:

Authentication attempts to ensure that an individual is who they claim to be. The need for this in an online transaction is obvious. Related to this is **nonrepudiation**, which deals with the ability to verify that a message has been sent and received and that the sender can be identified and verified.[4]

2.3.2 Cryptographic Algorithms

Cryptography is the science of encrypting, or hiding, information—something people have sought to do since they began using language. although language allowed people to communicate with one another, those in power attempted to hide information by controlling who was taught to read and write.

Eventually, more complicated methods of concealing information by shifting letters around to make the text unreadable were developed, these complicated methods are cryptographic algorithms, also known as ciphers, the word cipher comes from the Arabic word sifr, meaning empty or zero.[4]

Cryptography is a branch of mathematics based on the transformation of data, traditionally Cryptography is associated only with keeping data secret however, modern cryptography provides an important tool for protecting information and is used in many aspects of computer security, for example, cryptography can help provide data confidentiality, integrity, electronic signatures, and advanced user authentication. although modern cryptography relies upon advanced mathematics, users can reap its benefits without understanding its mathematical underpinnings.[5]

Cryptography relies upon two basic components: an algorithm (or cryptographic methodology) and a key, in modern cryptographic systems, algorithms are complex mathematical formulae and keys are strings of bits, for two parties to communicate, they must use the same algorithm (or algorithms that are designed to work together),

the algorithms stay the same in every implementation, but a different key is used for each, which ensures that even if someone knows the algorithm you use to protect your data, he cannot break your security, following illustration shows a diagram of the encryption and decryption process and its parts.[4]

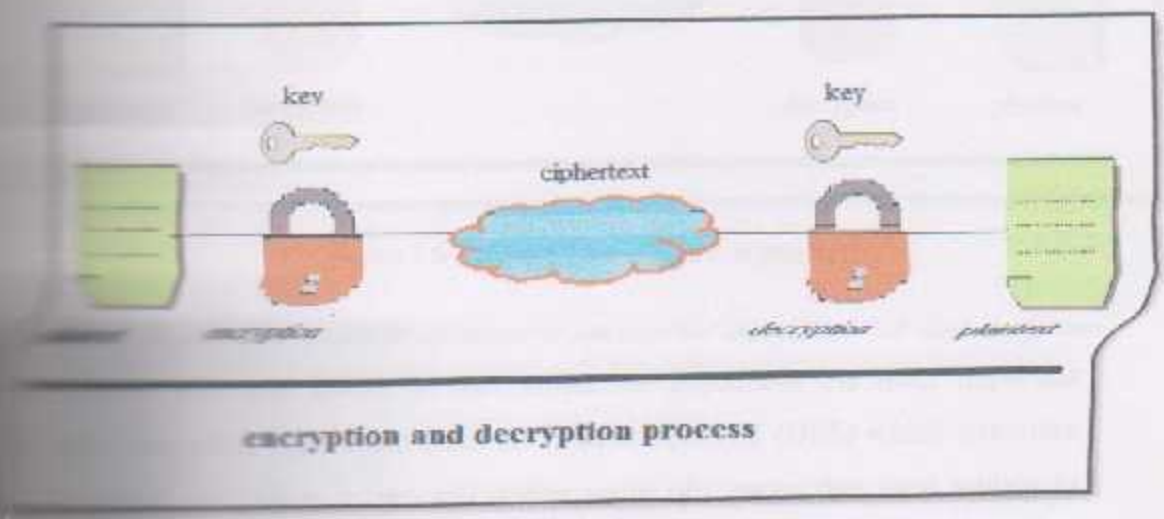

Figure 2.5 encryption and decryption process

There are three basic types of cryptography: symmetric systems (also called secret key systems), asymmetric systems (also called public key systems), and hashing one-way functions, each type is best suited for particular situations.[4]

Symmetric Encryption

Symmetric encryption is the older and more simple method of encrypting information. The basis of symmetric encryption is that both the sender and the receiver of the message have previously obtained the same key, all symmetric algorithms are based upon this shared secret principle.[4]

Figure 2.6 is a simple diagram showing the process, a symmetric algorithm goes through the process encryption from plaintext to ciphertext, this ciphertext message is, presumably, transmitted to the message recipient, who goes through the process to decrypt the message using the same key that was used to encrypt the message. Figure 2.7 shows the keys to the algorithm, which are the same value in the case of symmetric encryption.

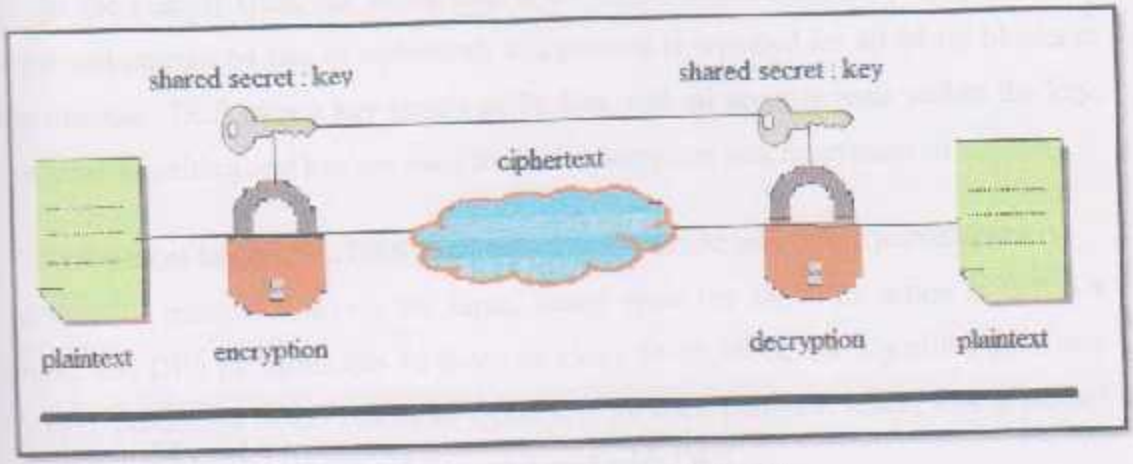

figure 2.6 Layout of a symmetric algorithm

Some of the more popular symmetric encryption algorithms in use today are DES, 3DES, AES, and IDEA, in this subsection introduces the most important symmetric encryption algorithms: Data Encryption Standard (DES) which simulates in this project, but before it, we will define some of terms that used widely in cryptographic algorithms.[3]

- Plain text: this is the original message or data that is fed into the algorithm as input.
- Encryption algorithm: the encryption algorithm performs various substitutions and transformations on the plaintext.
- Key: it is also input to the encryption algorithms, the exact substitution and transformations performed by the algorithm depend on the key.
- Ciphertext: this is the scrambled message produced as output, it depends on the plaintext and the key, for a given message, two different keys will produce two different ciphertext.
- Decryption algorithm: this is essentially the encryption algorithm run in reverse, it take the ciphertext and the key and produce the plaintext.

Data Encryption Standard (DES) :

The most widely used encryption scheme, it segments the input data into blocks of a specified size, typically padding the last block to make it a multiple of the block size required.

In the case of DES, the block size is 64 bits, which means DES takes a 64-bit input and outputs 64 bits of ciphertext, this process is repeated for all 64-bit blocks in the message, DES uses a key length of 56 bits, and all security rests within the key, the same algorithm and key are used for both encryption and decryption.[3]

At the most basic level, DES performs a substitution and then a permutation (a form of transposition) on the input, based upon the key, this action is called a round, and DES performs this 16 times on every 64-bit block, the algorithm goes step by step, producing 64-bit blocks of ciphertext for each plaintext block, this is carried on until the entire message has been encrypted with DES.[3]

As mentioned, the same algorithm and key are used to decrypt and encrypt with DES, the only difference is that the sequence of key permutations is used in reverse order.

Over the years that DES has been a cryptographic standard, a lot cryptanalysis has occurred, and while the algorithm has held up very well, some problems have been encountered, for example multiple successful attacks against DES algorithms have used fewer rounds than 16, any DES algorithm with fewer than 16 rounds could be analyzed more efficiently with chosen plaintext than via a brute_force attack using differential cryptanalysis, with 16 rounds and not using a weak key, DES is reasonably secure and, amazingly, has been for more than two decades.[3]

In 1998, a distributed effort consisting of a supercomputer and 100,000 PCs over the Internet was made to break a 56-bit DES key, by attempting more than 240 billion keys per second, the effort was able to retrieve the key in less than a day, this demonstrates an incredible resistance to cracking a 20-year-old algorithm, but it also demonstrates that more stringent algorithms are needed to protect data today.[4]

DES Asymmetric Encryption

Asymmetric encryption is in many ways completely different from symmetric cryptography, while both are used to keep data from being seen by unauthorized users, asymmetric cryptography uses two keys instead of one, also it based on mathematical functions rather than on simple operations on bit patterns such as are used in symmetric encryption algorithms, it was invented by Diffie and Hellman in

asymmetric cryptography is more commonly known as public key cryptography.[4]

The system uses a pair of keys: a private key that is kept secret, and a public key that can be sent to anyone, the system's security relies upon resistance to deducing one key, given the other, and thus retrieving the plaintext from the ciphertext.

RSA, Diffie-Hellman, elliptic curve cryptography (ECC), and ElGamal are all popular asymmetric algorithms, we briefly introduce the RSA algorithm which will continue in the second semester.[3]

RSA

RSA is one of the first public key cryptosystems ever invented, It can be used for both encryption and digital signatures, RSA is named after its inventors, Ron Rivest, Adi Shamir, and Leonard Adleman, and was first published in 1977.[3]

This algorithm uses the product of two very large prime numbers and works on the principle of difficulty in factoring such large numbers, it's best to choose large prime numbers that are from 100 to 200 digits in length and are equal in length, these two primes will be P and Q. Randomly choose an encryption key, E, so that E is greater than 1, E is less than $P * Q$, and E must be odd. E must also be relatively prime to $(P - 1)$ and $(Q - 1)$, then compute the decryption key D:[4]

$$D = E^{-1} \, mod \, ((P - 1)(Q - 1))$$

Now that the encryption key and decryption key have been generated, the two prime numbers can be discarded, but they should not be revealed, to encrypt a message, it should be divided into blocks less than the product of P and Q. Then,

$$Ci = Mi^{E} mod(P * Q)$$

C is the output block of ciphertext matching the block length of the input message M. To decrypt a message, take ciphertext, C, and use this function:

$$Mi = Ci^{D} mod(P * Q)$$

The use of the second key retrieves the plaintext of the message.

This is a simple function, but its security has withstood the test of more than 20 years of analysis. Considering the effectiveness of RSA's security and the ability to have two keys, why are symmetric encryption algorithms needed at all? The answer is

speed. RSA in software can be 100 times slower than DES, and in hardware it can be even slower.[4]

Since the security of RSA is based upon the supposed difficulty of factoring large numbers, the main weaknesses are in the implementations of the algorithms, until recently, RSA was a patented algorithm, but it was actual standard for many years.

2.3.2.2 Hash Function Algorithms

Hash functions are commonly used encryption methods, a hashing function or hash function is a special mathematical function that performs one-way encryption, which means that once the algorithm is processed, there is no feasible way to use the ciphertext to retrieve the plaintext that was used to generate it. Also, ideally, there is no feasible way to generate two different plaintexts that compute to the same hash value. The hash value is the output of the hashing algorithm for a specific input, the illustration shows the one way nature of these functions.[4]

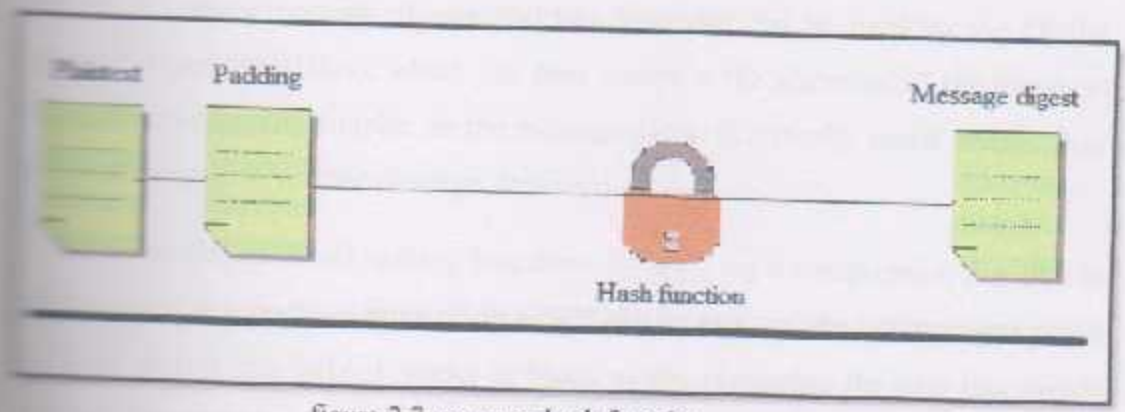

figure 2.7 one way hash function

Common uses of hashing functions are to store computer passwords and to ensure message integrity, This is the primary purpose for which the algorithms were designed, the idea is that hashing can produce a unique value that corresponds to the data entered, but the hash value is also reproducible by anyone else running the same algorithm against the same data. So you could hash a message to get a message authentication code (MAC), and the computational number of the message would show that no intermediary has modified the message, this process works because hashing algorithms are typically public, and anyone can hash data using the specified algorithm. It is computationally simple to generate the hash, so it is simple to check

the validity or integrity of something by matching the given hash to one that is locally generated. [4]

There are two popular hash algorithms are the Secure Hash Algorithm (SHA) series and Message Digest (MD) hash of varying versions (MD2, MD4, MD5), in this subsection SHA_1 will be introduced, to simulate in the second semester.

SHA

Secure Hash Algorithm (SHA) refers to a set of four hash algorithms. these algorithms are included in the SHA standard Federal Information Processing Standards (FIPS) 180-2, the individual standards are named SHA-1, SHA-256, SHA-384, and SHA-512, the latter three variants are occasionally referred to collectively as SHA-2.[3]

SHA_1

SHA-1, developed in 1993, was designed as the algorithm to be used for secure hashing, it creates message digests 160 bits long that can be used by the Digital Signature Algorithm (DSA), which can then compute the signature of the message. This is computationally simpler, as the message digest is typically much smaller than the actual message—smaller message, less work.[4]

SHA-1 works, as do all hashing functions, by applying a compression function to the data input. It accepts an input of up to 2^{64} bits or less and then compresses down to a hash of 160 bits. SHA-1 works in block mode, separating the data into words first, and then grouping the words into blocks.

The words are 32-bit strings converted to hex; grouped together as 16 words, they make up a 512-bit block. If the data that is input to SHA-1 is not a multiple of 512, the message is padded with zeros and an integer describing the original length of the message.[4]

Once the message has been formatted for processing, the actual hash can be generated, the 512-bit blocks are taken in order until the entire message has been processed, the computation uses eighty 32-bit words labeled $W0$, $W1$, $W2$, ..., $W79$ being sent to two, five-word buffers. The first five-word buffer's words are labeled A, B, C, D, E, and the second five-word buffer's words are labeled $H0$, $H1$, $H2$, $H3$, and

These buffers are combined until all words have been processed through all blocks of the message, and the entire message is then represented by the 160-bit string *H0 H1 H2 H3 H4*.[4]

At one time, SHA-1 was one of the more secure hash functions, but it has been found to be vulnerable to a collision attack, thus, many security professionals are suggesting that implementations of SHA-1 be moved to one of the other SHA versions. These longer versions, SHA-256, SHA-384, and SHA-512, all have longer hash results, making them more difficult to attack successfully. The added security and resistance to attack in SHA-2 does require more processing power to compute the hash.[3]

2.4 Using Multimedia In Education

The world in which we live is changing rapidly and the field of education is experiencing these changes in particular as it applies to Media Services. The old days of an educational institution having an isolated audio-visual department are long gone! The growth in use of multimedia within the education sector has accelerated in recent years, and looks set for continued expansion in the future.[6]

Teaching primarily requires access to learning resources, which can support concept development by learners in a variety of ways to meet individual learning needs, the development of multimedia technologies for learning offers new ways in which learning can take place in universities, schools, and the homes.

Definition: Multimedia combines five basic types of media into the learning environment: text, video, sound, graphics and animation. Multimedia simply combines these elements, thus providing a powerful new tool for education.[6]

A multimedia Learning environment involves a number of components or elements in order to enable learning to take place. Hardware and software are only part of the requirement. As mentioned earlier, multimedia learning integrates five types of media to provide flexibility in expressing the creativity of a learner .[6]

- *Text: out of all of the elements, text has the most impact on the quality of the multimedia interaction. Generally, text provides the important information.* Text acts as the keystone tying all of the other media elements together. It is well written text that makes a multimedia communication wonderful.
- Sound: Sound is used to provide emphasis or highlight a transition from one page to another. Sound synchronized to screen display, enables teachers to present lots of information at once. This approach is used in a variety of ways, [illegible] explanation.
- Video: The representation of information by using the visualization capabilities of video can be immediate and powerful, video provides new and exciting possibilities for the use of it in education, video help in placing a theoretical concept into context, also video can be used to tell readers what to do next, one of the most compelling justifications for video may be its dramatic ability to elicit an emotional response from the learner, such a reaction can provide a strong motivational to choose and persist in a task. The use of video is appropriate to convey information about environments that can be either dangerous or too costly to consider, or recreate, in real life.
- Animation: is used to show changes in state over time, or to present information slowly to learner so they have time to assimilate it in smaller chunks. Animations, when combined with user input, enable learners to view different versions of change over time depending on different variables. Animations are primarily used to demonstrate an idea or illustrate a concept, animations are based on drawings, that appear to move to illustrate a point.
- Graphics: they provide the most creative possibilities for a learning session. They can be photographs, drawings, graphs, that are using different range of skills: color, form, line, dimension, texture, visual Rhythm, and especially imagination.

[illegible] multimedia in education offers many benefits including:[8]

- Provide learners with opportunities to represent and express their prior knowledge.

- Allow learners to function as designers, using tools for analyzing the world, accessing and interpreting information, organizing their personal knowledge, and representing what they know to others.
- Encourages deep reflective thinking .

There are multimedia program, that will be used in this project, which is Silverlight program.

Using Silverlight in education:

Microsoft Silverlight is a powerful tool for creating and delivering rich Internet applications and media experiences, Silverlight introduces more than 40 new features, including dramatic video quality and performance improvements as well as features that improve developer productivity, it is freely available and supported large number of all platforms and browsers.

As mentioned, Silverlight create 'rich' content that encourages active learning, it allows learners to interact with the problem they are trying to solve, integrated animation and sound creates learning environment that is interesting and funny, a combination of problem_solving and fun equals a good tool for learning.

3

Chapter Three

Project Conceptual Design

3.1 Introduction

3.2 System Function

3.3 Software Tools Used

3.4 System Analysis

3.5 How System Work

3.6 Summery

3.1 Introduction

In a time when both highly personal and classified information is flowing through various public and private networks, and a time when identity theft has become more common, people are necessarily concerned about the security of data. So individuals employ numerous techniques- such as passwords, firewalls, and authentication- to guarantee the safety of the hardware, software, and data of their computer systems.

While all of these techniques are important probably the most important security procedure in a public network is the use of encryption, encryption is the basis for most aspects of secure communication. Since encryption is such an important topic, it is necessary that computer engineering and science students learn something about it.

However, since encryption algorithms are by necessity, extremely complex, and since the core curriculum of most computer science programs is already packed full of foundational computer science concepts, the question is how can the subject of encryption be incorporated into a computer science program?

While there are several courses where studying encryption is appropriate, the problem of finding an easy method to make these complex algorithms more understandable is not as a simple. In class, instructors often use teaching aids of some sort to help the students learn. The challenge encountered when designing a teaching tool for encryption, however, is that most modern encryption algorithms operate on the bits of a message rather than the characters in the message, in addition these algorithms are extremely complex.

These two factors imply that massive amounts of data must be shown to the student in order to explain any particular algorithm, this generally requires the teacher to pick a small example to go over in class – therefore not allowing much repetition or "what if analysis" on the part of the student. Due to the dynamic nature of students' questions and the overwhelming amount of information that must be presented in order to effectively describe an encryption algorithm it is difficult to integrate

encryption algorithms into the computer science curriculum. Therefore, this project provides an interactive learning tools that professors can use for demonstration purposes and that students can use for gain purposes.

3.2 System Function:

Encryption algorithms are by necessity extremely complex, this project aims to make the encryption algorithms, more understandable for computer engineering and computer science students, by providing new educational tool that provides interactive step by step demonstrations of the encryption processes for various algorithms. The main objectives of this project are:

- Identifying some of the security tools and algorithms.
- Using simulation to show and understand how the security algorithms work.
- Designing interactive environment for cryptographic tools and algorithms, to use with face to face learning or self reading material.
- Making the tool inexpensive or freely available.

3.3 Software Tools Used

In this project we need the following software tools to be used:

- Multimedia program, Silverlight tool, which is powerful and flexible tool , for creating rich applications and media, it will use in this project to build interactive animation for cryptography algorithms.
- Using programming language C# in Microsoft's Visual Studio 2010.NET.

3.4 System Analysis

Since this project acts pedagogical tool that provides interactive step by step demonstrations of the encryption processes for various cryptography algorithms and tools, and as we mentioned, when this project will design, it aims to achieve all domains of Bloom's taxonomy, which are Knowledge, Skills, and Attitude.

3.4.1 Intended Outcomes

the intended outcomes that we expected that the leaner who will use this learning environment gains from this project are:

- In knowledge domain:

    1. Identifying the basic types of cryptography algorithms and tools.
    2. List the most common algorithms in use for each type.
    3. Describe and explain these algorithms, and how them work.
    4. Compare and Evaluate these algorithms.

- In attitude domain:

    1. ability to select the appropriate algorithms according to specific situation.
    2. Differentiate between these algorithms.

- In skills domain:

    1. Responding and interactive with the encryption algorithms and tools.
    2. Practice and manipulate these algorithms.
    3. Using these algorithms in his or her life.
    4. Building and construct a similar or new algorithms.

3.4.2 Project Requirements:

To achieve these intended outcomes, the interactive learning environment that will design in this project, should have several different requirement.

1. It should have information about the cryptographic algorithms and tools, that describe these algorithms and how these work.
2. It should allow the user to select from a variety of algorithms.
3. The user should be able to define the encryption key when *appropriate*.
4. The user should also be able to define the string being encrypted.
5. The user should be able to receive interactive tutoring on more difficult concepts within an algorithm including a step by step explanation of the algorithm.
6. It should easily lend itself to being modified to include other algorithms this implies that all essential code be open source.
7. The program should be inexpensive or free.

3.4.3 Use-Cases Of Project

After we describe the requirements of the project, we will talk about the basic use-cases, that describe the interaction between the user and the system, there are many use-cases, and two actors, the system administrator and system user.

The system administrator will design the simulator, so he will program, configure, test the simulator, while the user of system interact with simulator functions, as run the simulator, choose specific algorithms, enter required information and request events to do.

The following figure depicts a preliminary use-case diagram for the simulator, each use-case is represented by an oval.

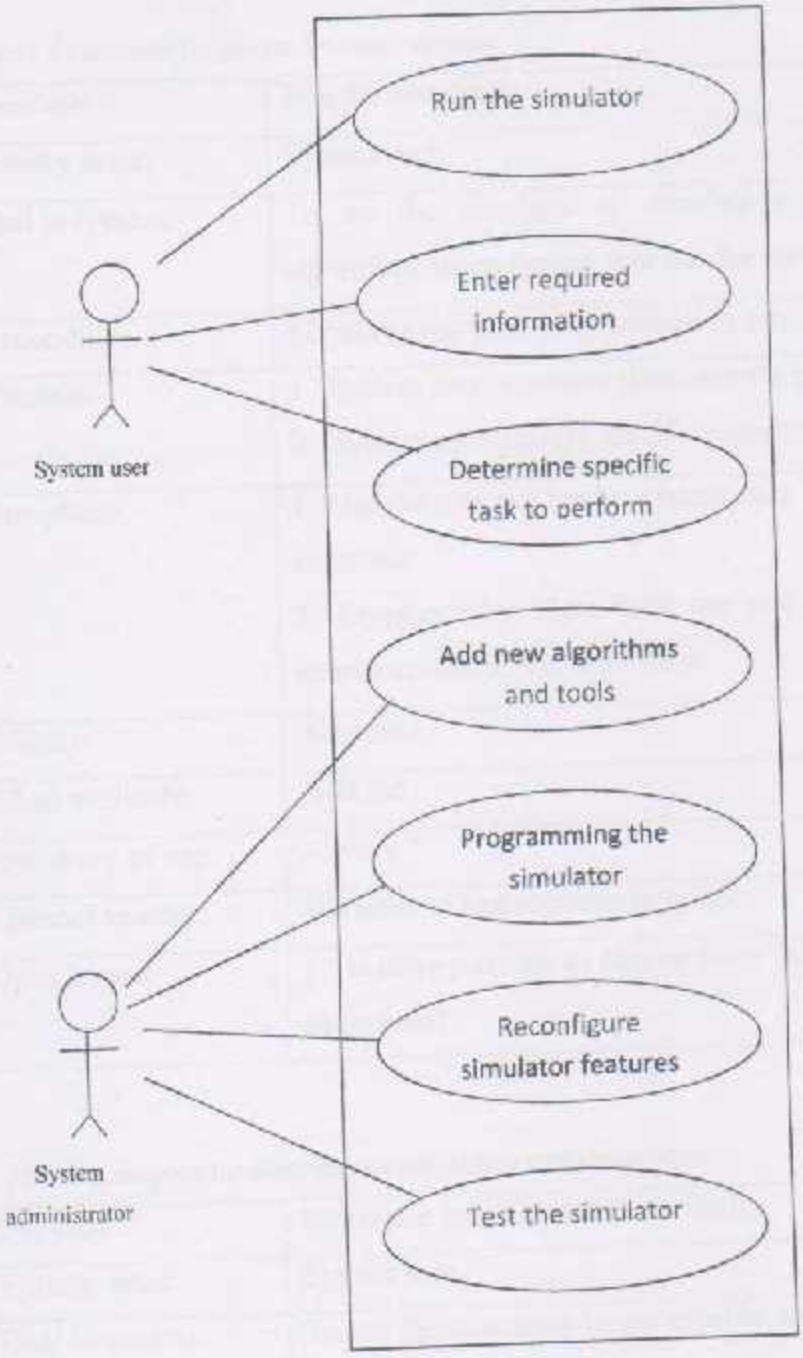

Figure 3.1 preliminary use-case diagram for interactive security algorithms simulator

The following tables shows use-cases template for detailed description of use-cases.

Table 3.1 template for run the simulator use-case.

Use-case:	Run the simulator.
Primary actor:	System user.
Goal in context:	To set the simulator to simulate some of cryptography algorithms when system user run the simulator.
Preconditions:	Simulator has been programmed to simulate the algorithms.
Scenario:	1. System user: accesses simulator via internet. 2. System user: choose specific algorithm.
Exceptions:	1. Simulator is not ready: system user checks the domain of simulator. 2. Cryptography algorithms are not chosen: system user checks to choose the algorithm.
Priority:	Essential.
When available:	First run.
Frequency of use:	Always.
Channel to actor:	Via internet and software interface.
Open issues:	1. Is there possible to choose more than one algorithm at the same time?

Table 3.2 template for determine specific task to perform use-case.

Use-case :	Determine specific task to perform.
Primary actor:	System user.
Goal in context:	To set the simulator to do specific tasks, that chosen by the system user.
Preconditions:	Simulator has been programmed to do specific tasks.
Scenario:	1. System user: choose about specific algorithm. 2. System user: choose full encryption. 3. System user: choose step by step encryption.
Exceptions:	1. System user doesn't determine specific task to do by the simulator: system user should choose task to perform.

Priority:	Not essential.
When available:	When run the simulator.
Frequency of use:	Always.
Channel to actor:	Via internet and software interface.
Open issues:	1. Is there possible to choose more than one task to perform at the same time? 2. should the simulator perform additional task?

Table 3.3 template for enter required information use-case.

Use-case :	Enter required information.
Primary actor:	System user.
Goal in context:	To set the simulator to simulate some of cryptography algorithms .
Preconditions:	Simulator has been programmed to require the needed information, and manipulated with wrong information.
Scenario:	1. System user: choose specific algorithm. 2. System user: enter the correct information. 3. System user: deal with messages boxes.
Exceptions:	1. the entered information is wrong: system user reenter the correct information.
Priority:	Essential, must be implemented.
When available:	First run.
Frequency of use:	Always.
Channel to actor:	Via internet and software interface.
Open issues:	1. How much time does the user system have to enter the information from the first time enter the information? 2. Is there a way to correct the wrong information that entered by the system user?

Table 3.4 template for programming the simulator use-case.

Use-case :	Programming the simulator.
Primary actor:	System administrator.
Goal in context:	To set the simulator to simulate some of cryptography algorithms.
Preconditions:	Prepared the software tools used.
Scenario:	1. System administrator: create the classes that have the algorithm's methods and members. 2. System administrator: create the user interface for that algorithms.
Priority:	Essential, must be implemented.
Channel to actor:	software tools used (VS 2010.NET).
Open issues:	1. Is there another mechanisms and software tools can be used to program the simulator?

Table 3.5 template for add new cryptography algorithms use-case.

Use-case :	Add new cryptography algorithms.
Primary actor:	System administrator.
Goal in context:	To set the simulator to simulate the most important cryptography algorithms.
Preconditions:	Simulator has been programmed to make each algorithm a separate class, and separated from the user interface.
Scenario:	1. System administrator: create the class that has the algorithm's methods and members. 2. System administrator: create the interface for that algorithm.
Priority:	Essential, must be implemented.
When available:	When run the simulator.
Frequency of use:	Always.
Channel to actor:	Via software tools that used.
Open issues:	1. Is there possible to allow the system user to add new cryptography algorithms?

Table 3.6 template for reconfigure simulator features use-case.

Use-case :	Reconfigure simulator features.
Primary actor:	System administrator.
Goal in context:	*To improve the simulator to support another features.*
Preconditions:	Simulator has been programmed to be readily modified and make each algorithm a separate class, and separated from the user interface.
Scenario:	1. System administrator: reconfigure the demonstration of the algorithms. 2. System administrator: reconfigure any part of user interface.
Priority:	Not essential.
When available:	When run the simulator.
Frequency of use:	Always.
Channel to actor:	Via software tools and software interface.
Open issues:	1. Is there possible to allow to the user system to reconfigure simulator features?

Table 3.7 template for test the simulator use-case.

Use-case :	Test the simulator.
Primary actor:	System administrator.
Goal in context:	To be sure that simulator run successfully and it achieve the expected goals.
Preconditions:	Simulator has been programmed to achieve the expected goals, and it was used by users.
Scenario:	1. System user: uses the simulator, and answers a collection of answers. 2. System administrator: test the written code and analyses the answers of system users.
Priority:	Essential, must be implemented.
When available:	After the simulator programmed and using by the users.
Frequency of use:	First run.

Channel to actor:	Via simulator and software interface.
Open issues:	1. How can be sure from the correctness of user's answered?

3.5 How System Work

This section talk about the mechanisms that will be followed to design the simulator in order to achieve the project functions with project requirements.

So to meet all of above specifications, this project provides tool that contains the previous features, so when this simulator design, it designs each algorithm a separate class which contain the algorithm's data members and methods, these classes created in a separate class library from the user interface, to ensure the ease of adding future features.

After creating the class, then we create the interface for each algorithm, the interface will be form-based, this design will allow to inherit the forms appearance from the other forms in order to create new form, also the user will control progress through the algorithm through the use of several buttons labeled with the steps that will take place.

When the users use this simulator, many windows will appears, which will allow the users to interactive with it, by several buttons labels with steps, this interactive allowing the user to select the algorithms, define the encryption key, and define the string being encrypted, so during the process of setting up the algorithm, the simulator checks all user input to ensure proper values are entered.

Since this project using Silverlight tool, the simulator view demonstrations of certain steps within the algorithm, for each demonstration the appropriate graphical and animation will use, all of the demonstrations, like the simulator, are user interactive allowing the user to step through the demonstration until completion of the particular step.

3.6 Summery

This project design an easily modifiable, interactive simulator for teaching encryption algorithms, this simulator supports multiple algorithms, allows the user to define the encryption key when appropriate, allows the user to define the string and the size of the string being encrypted, provides interactive demonstrations of components within an algorithm by using multimedia program, and provides interactive step by step explanations of the algorithms, furthermore, made the simulator freely available.

4

Chapter Four

Software System Design

4.1 Introduction

4.2 General Design Option

4.3 Software Detailed Description Of System Components

4.1 Introduction

This chapter presents project software in more details, it presents design options and tools that used in this project, and mechanisms used to implement the project and achieve the goal of it.

4.2 General Design Option

4.2.1 Software Application Options

Since this project acts interactive tool for learning some of encryption algorithms, and it uses bloom's taxonomy of learning, so there are many languages can use as Java, Vb, C#, and many programs to support multimedia as Adobe flash, Silverlight.

We prefer to use .NET environment to program this project, so we use Visual Studio 2010.NET which consider as one of famous tools that used to develop applications in .NET environment, we use C# language in Visual Studio 2010

- Programmer Reasons

  The programmer of the system has a good background knowledge and good experience that can help to install, configure and deal with Visual Studio 2010 in a good way and use C# language rather than other languages.

- Technology Used

  Using .NET environment since it acts new technology represents basic to develop applications that can connect systems and information and devices and users in one and specialized frame.

- Tools Used

  Using Visual Studio 2010 which is a powerful software used to develop many applications, and using Microsoft Silverlight since it can work in any operating systems and any browsers and can work on it easily.

- Application Libraries and good user guide

  Visual Studio 2010 has a huge build-in libraries in it, and they are open source, so they can help the programmer for develop system in easily way, and also support user guide file that help the programmers to use application and gives them the necessary assistant while programming process.

4.2.2 Environment Application

This simulator can work as web application or desktop application, since we use Silverlight and Silverlight is web application it will work as web application.

4.2.3 Software Requirement Specification

This section describes the software program that used in this project

4.2.3.1 Visual Studio 2010.NET

Visual studio 2010 is a powerful integrated development environment (IDE) that ensures quality code throughout the entire application life cycle from design to deployment. Whether developing applications for SharePoint, the web, Windows, Windows Phone, and beyond.

To program this project in Visual studio 2010, need some requirements to work properly:

- Software requirement :

  Visual studio 2010 can be installed on the following operating systems: Windows XP, Windows Vista, Windows 7, Windows Server.

- Hardware requirement:

  - Computer that has a 1.6 GHz or faster processor.
  - 1G (32 bit) or 2G (64 bit).
  - 3GB of available hard disk space.

As mentioned this project uses multimedia programs in learning, and we choose Silverlight program to perform this task, Visual studio 2010 is one of the best tools that used to develop Silverlight.

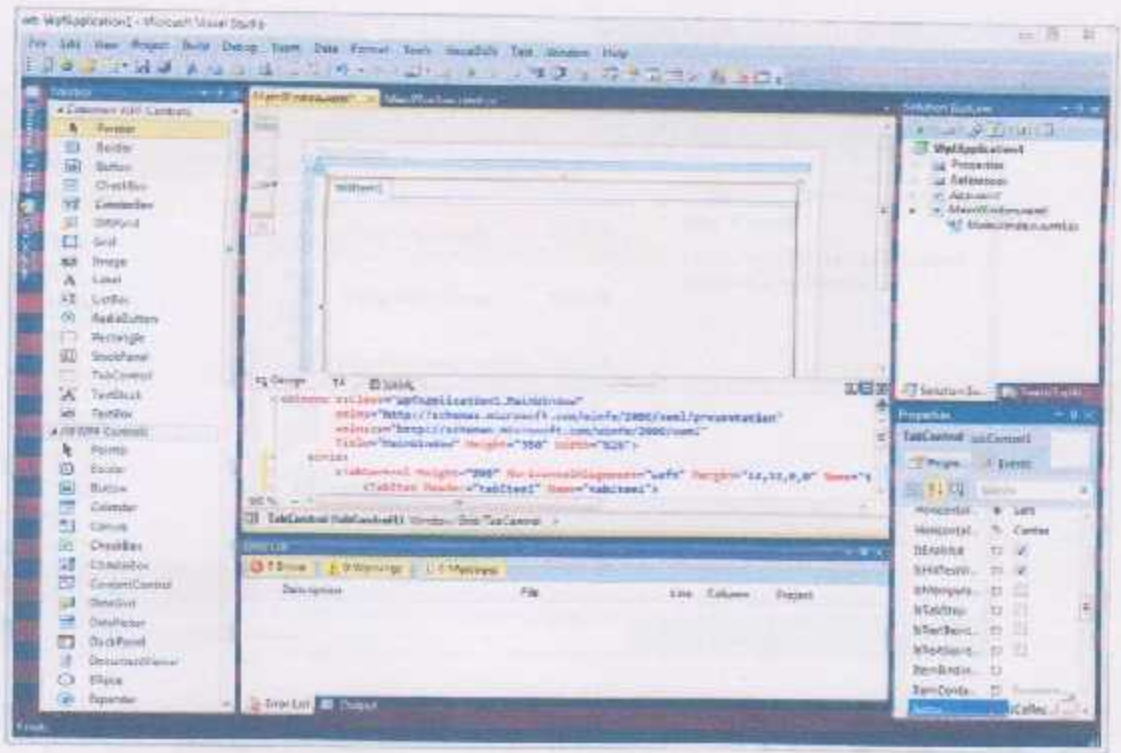

Figure 4.1 Visual Studio 2010 .NET environment

4.2.3.2 Microsoft Silverlight 4

Microsoft Silverlight is a powerful tool creating and delivering rich internet applications (RIA) and media experiences on the web.

To use Silverlight in programming, one of tools which used to develop it should be installed in our project use Visual Studio 2010, and Silverlight also need some requirements to work properly:

- Silverlight 4 SDK : which act the basic framework.
- Silverlight 4 tools for Visual Studio 2010.
- Silverlight 4 toolkit.

In this section, describe how build Silverlight applications, and the files that will produce, as show in figure 4.2 we choose Silverlight application in Visual C# to build Silverlight application by using Visual Studio 2010.NET.

Figure 4.2 Creation Silverlight application

After we creation the application, two projects will produce as shown in figure 4.3 in solution explorer, the two projects are:

- Silverlight application : which is Silverlight project.
- Silverlight application.Web : which contains the web page, where Silverlight works in it.

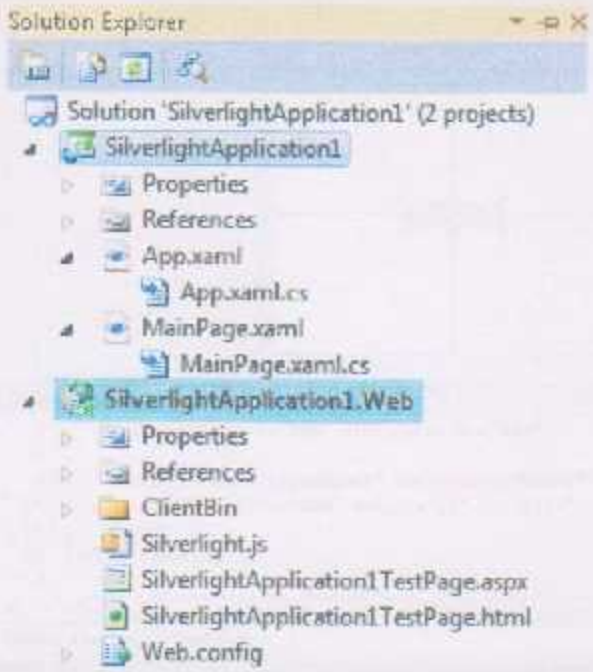

Figure 4.3 solution explorer in Silverlight

In web project there are two pages:

- Silverlight application test page.aspx
- Silverlight application test page.html

In Silverlight project there are many files, two of them are:

- App.xaml: connected with it App.xaml.cs, which contain the code that will be run.

- MainPage.xaml: connected with MainPage.xaml.cs which acts the main form for the application.

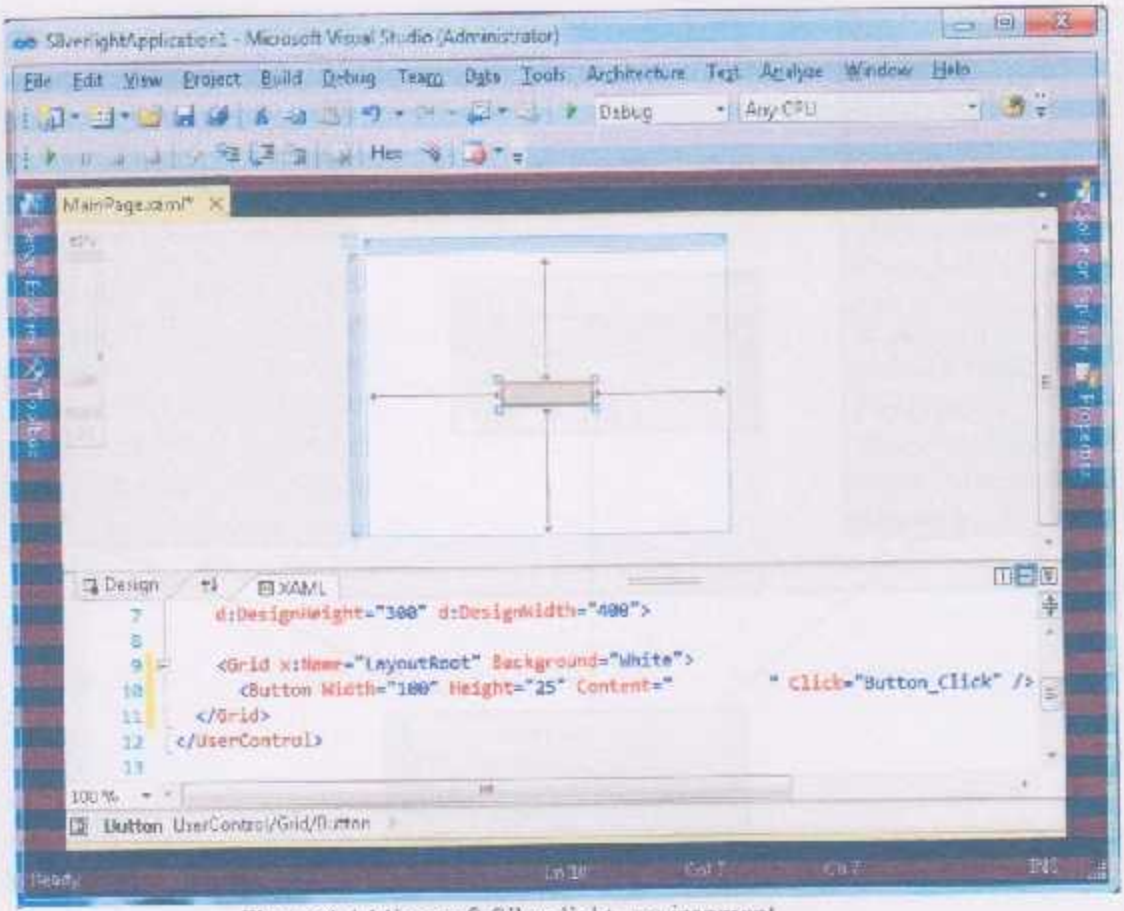

Figure 4.4 Microsoft Silverlight environment

4.3 Software Detailed Description Of System Components

This section describe the system design in more details, it describes the classes of the project, and the methods and members of each class.

To achieve the most separation between code and GUI, the majority of code is written in separated classes and an instances of these classes are created, the project has four basic classes.

1. DES class: it is responsible about simulate DES algorithm.

2. RSA class: it is responsible about simulate RSA algorithm.

3. SHA_1 class: it is responsible about simulate SHA_1 algorithm.

4. user interface class.

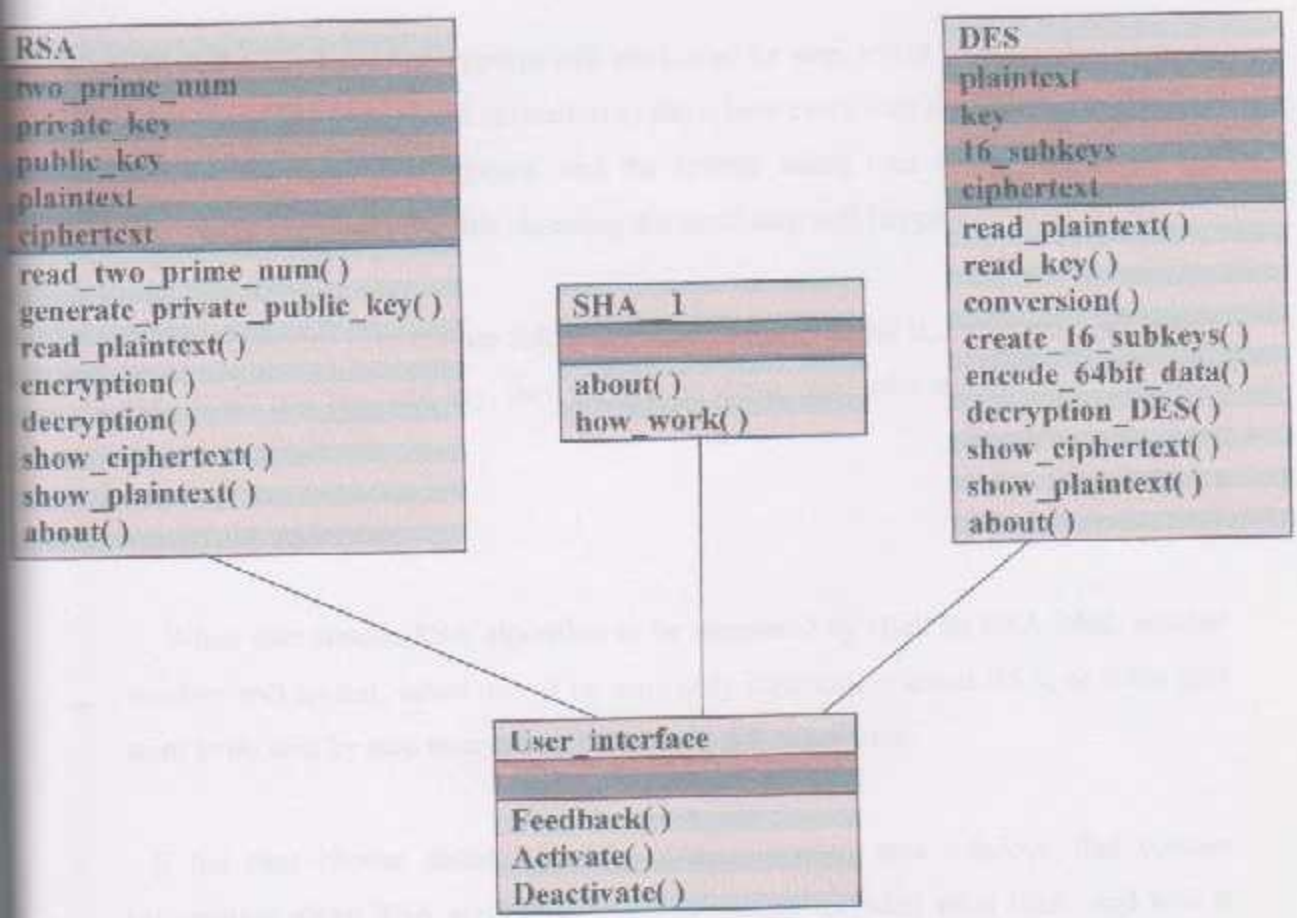

Figure 4.5 design class for simulator

4.3.1 Data Encryption Standard (DES) Algorithm

When user choose DES algorithm to be simulated by click on DES label, another window will appear, asked user if he want only information about DES, or if the user want to do step by step encryption, or he want full encryption.

If the user choose about DES, the system opened new window, that contain information about DES algorithm, this information included what DES, and how it works, and the input and output of DES algorithm.

The second choose for the user is choosing step by step encryption, when user choose, he will interact with this algorithm, the system asked the user to enter the plaintext (the text that will be encryption), and to enter the key that will used in encryption process.

Then the process of encryption will start, step by step, every step appears alone in window, using graphics and animation to show how every step doing, after encryption finished, the ciphertext appears, and the system asked user if he want to do the decryption, if user choose this choosing the same step will happened but in reverse.

The third choose is to choose full encryption, if user choose it, the system asked him to enter the plaintext and key, then the system shows the ciphertext.

4.3.2 RSA Algorithm

When user choose RSA algorithm to be simulated by click on RSA label, another window will appear, asked user if he want only information about RSA, or if the user want to do step by step encryption, or he want full encryption.

If the user choose about RSA, the system opened new window, that contain information about RSA algorithm, this information included what RSA, and how it works, and how generate private and public keys, and the input and output of RSA algorithm.

The second choose for the user is choosing step by step encryption, when user choose, he will interact with this algorithm, the system asked the user to enter two different prime numbers to show how generation the private and public keys, plaintext (the text that will be encryption).

Then the process of encryption will start, step by step, every step appears alone in window, using graphics and animation to show how every step doing, after encryption finished, the ciphertext appears, and the system asked user if he want to do the decryption, if user choose this choosing the decryption will start.

The third choose is to choose full encryption, if user choose it, the system asked him to enter the plaintext and private key to do encryption. then the system shows the ciphertext.

4.3.3 Secure Hash Algorithm _1 (SHA_1)

When user choose SHA_1 algorithm to be simulated by click on SHA_1 label, another window will appear, asked user if he want only information about SHA_1, or if the user want to show how SHA_1 work.

If the user choose about SHA_1, the system opened new window, that contain information about SHA_1, this information included what SHA_1, and how it works, and the input and output of SHA_1.

The second choose for the user is choosing how SHA_1 work, if user choose this, the system will not asked user to enter the message that will be hashed since it very large reach 2^64 bit and it performs very large number of process, it will show how SHA_1 work step by step on small data by using graphic and animation.

4.3.4 User Interface

We designed simple and easy user interface, as mentioned this project was designed as web site, when user enter to this web site, three label will find, every one of this label represent one of the algorithm that will be simulated in this system, which that are : DES, RSA, SHA_1, figure 4.5 shows the label represent RSA algorithm.

Figure 4.6 RSA label

To choose one of algorithm, the user will click on the label that represent this algorithm, then the processes will start, uses message boxes dialog to interactive with user, to enter the required data, and to show if there any error happened.

5

Chapter Five

Implementation and Testing

5.1 Introduction

The whole testing stage will be described in this chapter, we test methods and procedures that made to implement the project, and test the outcomes of simulator to be sure if the project has achieved the expected goals.

5.2 Implementation And Testing Procedure

System testing is an important step in system implementation, it measures the effectiveness of the system before introducing it to the users.

As mentioned the project simulate three of the most common algorithms in computer security, this algorithms are DES, RSA, SHA_1, in project implementation every algorithm build in separate class, and every one of algorithm has it's programming, data input, outcomes that different from another algorithm.

In testing stage, first test the methods and code that written to simulate the algorithms, to be sure that gives a correct result, we compare the given answers from the simulator by the answers that notice from build-in library in visual studio 2010.NET called " system.security.cryptography" and notice the same answers, after we test the program and notice that run successfully, and since this project acts learning environment, that apply bloom's taxonomy we test if this simulator actually achieve this goals.

5.2.1 DES Implementation and Testing

The code that written for simulate DES algorithm runs successfully, the DES simulator accepted correct data, and deal with incorrect data, it asked user to inter correct date again or sometimes ignore it, then it process it step by step, first generate Create 16 sub keys, each of which is 48-bits long, second Encode each 64-bit block of data, finally output the correct result as shown in figure 5.1.

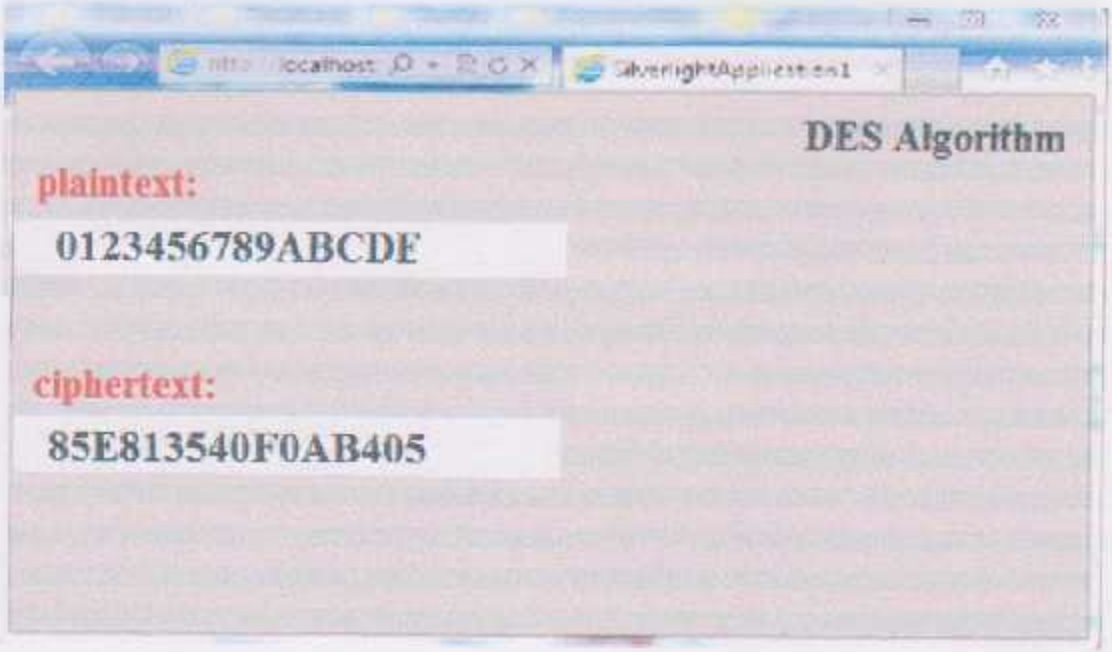

Figure 5.1 DES simulator

5.2.2 RSA Implementation and Testing

The code that written for simulate RSA algorithm runs successfully, the RSA simulator accepted correct data, and deal with incorrect data, it asked user to inter correct date again or sometimes ignore it, then it process it step by step, first generate the private key and public key, second encryption and decryption the data that user enter to the simulator, finally output the correct result. figure 5.2 shows step of generation private key and public key.

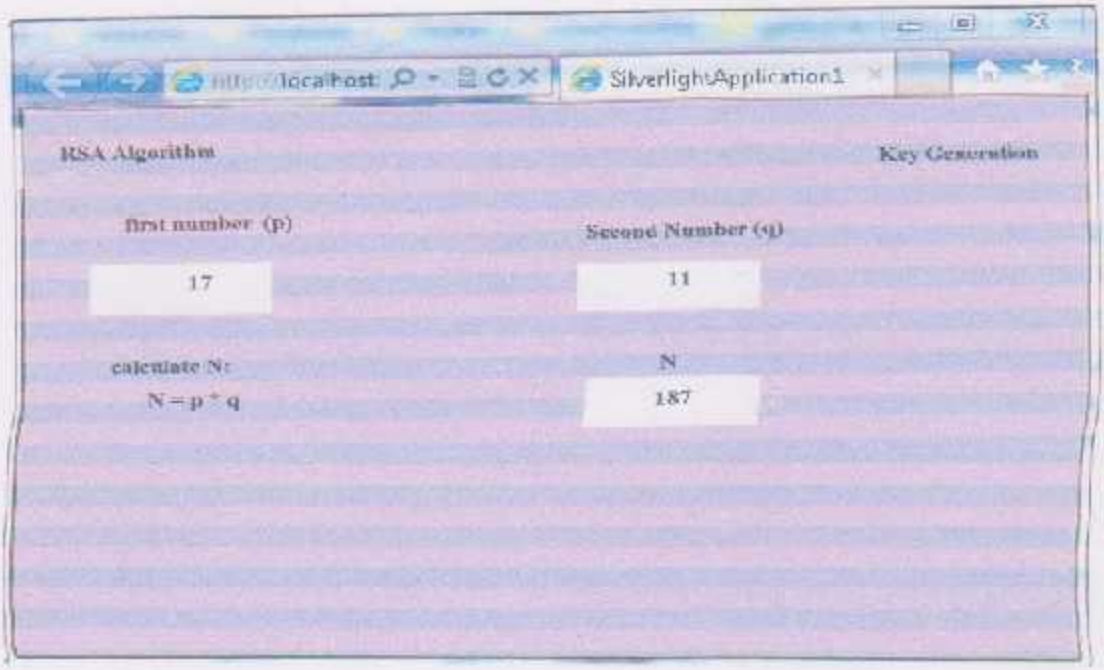

Figure 5.2 RSA simulator

5.2.3 SHA_1 Implementation and Testing

The code that written for simulate SHA_1 algorithm runs successfully, the SHA_1 simulator hashed samples of data since it requires a large amount of data, , then it process it step by step, first append padding bit, second process message in 512 bit blocks, finally output 512_bit message digest. figure 5.3 shows step of append padding and process massage in 512_bit blocks.

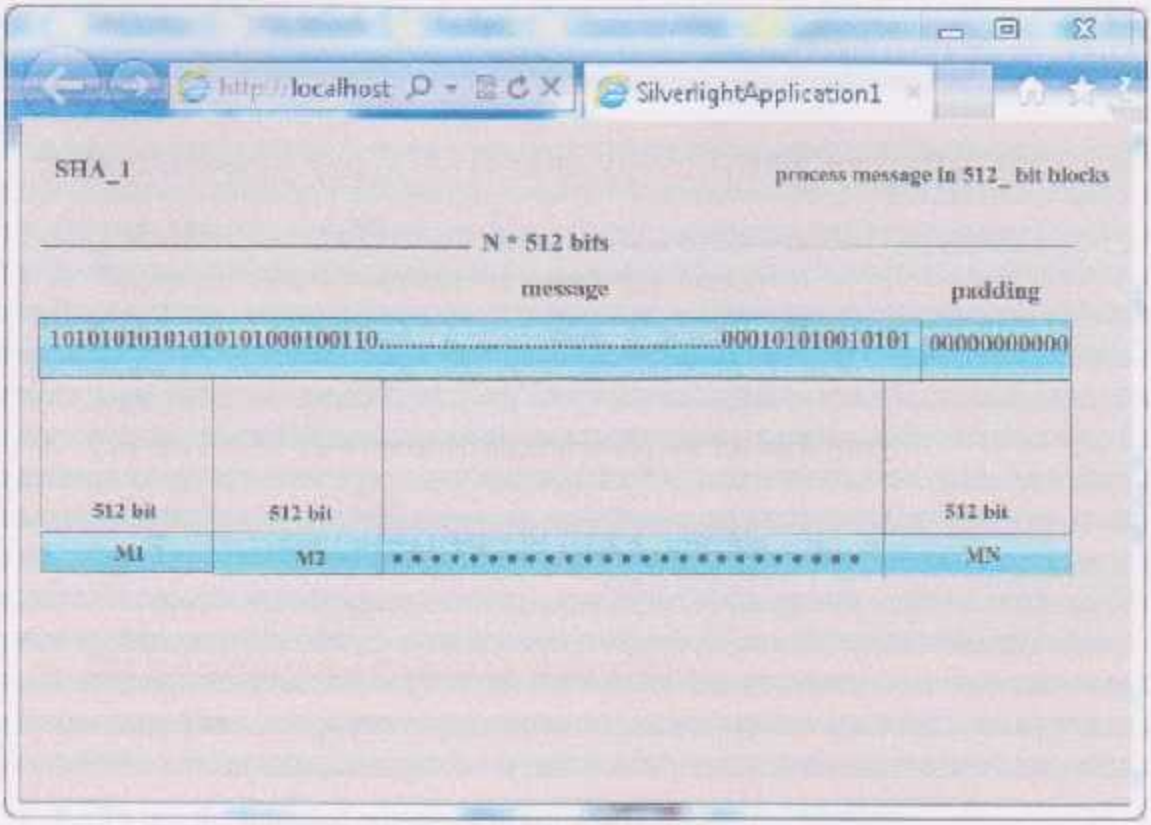

Figure 5.3 SHA_1 simulator

5.2.4 Outcomes Testing

Since this project acts pedagogical tool that provides interactive step by step demonstrations of the encryption processes for various cryptography algorithms and tools, and as we mentioned, when this project design, it aims to achieve all domains of bloom's taxonomy, which are knowledge, skills, and attitude.

In this section we will test if the intended outcomes that we expected that the user who will use this learning environment gains from this project actually achieves all domains of bloom's taxonomy.

So this simulator was be showing for a group of students in university, some of them study computer engineering or computer science, and some of them study another specialization, some of students who study computer engineering have a background of this algorithms, after they using this simulator, they answered a sum of questions, which are:

1. Do you study at any field of computer?

   ○ Yes ○ No

2. Do you have any background about this algorithms(DES, RSA, SHA_1)?

   ○ Yes ○ NO

➢ To test the intended outcomes in knowledge domain:

3. Can you identify the basic types of cryptography algorithms and tools?

   ○ Yes ○ No

4. Can you list the most common algorithms in use for each type?

   ○ Yes ○ No

5. Can you describe and explain these algorithms, and how them work?

   ○ Yes ○ No

6. Can you compare and evaluate these algorithms?

   ○ Yes ○ No

➢ To test the intended outcomes in attitude domain:

7. Can you able to select the appropriate algorithms according to specific situation?

   ○ Yes ○ No

8. Can you differentiate between these algorithms?

   ○ Yes ○ No

➢ To test the intended outcomes in skills domain:

9. Can you interactive with these encryption algorithms ?

   ○ Yes ○ No

10. Can you practice and manipulate these algorithms?

    ○ Yes ○ No

11. Can you use these algorithms in your life?

    ○ Yes ○ No

12. Can you build and construct a similar or new algorithms?

    ○ Yes ○ No

13. Please mail me if you find errors, I will appreciate you much.

    bayanihshish@yahoo.com

Also since this project build as web site, the above questions written in web site, and asked users to answered them, then the answers send to my email.

Analysis the questions:

After collected the questions and answers, and analyses them, we find that 24 students answered these questions, from those students 16 students study computer engineering or computer science, from those students only 7 students have background for these algorithms, and 8 student study another specialization.

For the second collection of questions (3 -6):

- 23 students answered they can identify the basic types of cryptography algorithms.

  96 % of students can identify the basic types of cryptography algorithms.
- 22 students answered they can list the most common algorithms in use for each type.

  92 % of students can list the most common algorithms in use.
- 18 students answered they can describe and explain these algorithms, and how them work.

  75 % of students can describe and explain these algorithms.
- 14 students answered they can compare and evaluate these algorithms

  58 % of students can compare and evaluate these algorithms.

For the third collection of questions (7-8):

- 16 students answered they can able to select the appropriate algorithms according to specific situation.

  67 % of students can able to select the appropriate algorithms.
- 17 students answered they can differentiate between these algorithms.

  71 % of students can differentiate between these algorithms.

For the fourth collection of questions (9-12):

- 14 students answered they can interactive with these encryption algorithms.

  58 % of students can interactive with these encryption algorithms.
- 11 students answered they can practice and manipulate these algorithms.

  46 % of students can practice and manipulate these algorithms.

- 4 students answered they can use these algorithms in his or her life.
  17 % of students can use these algorithms in his or her life.
- 1 students answered they can build and construct a similar or new algorithms.
  4 % of students can build and construct a similar or new algorithms.

We notice that results affected by numbers of things, from this things, the specialized of the learner, and the learner background, we notice that if the user study computer engineering or computer science or he has background of this algorithms, he will gain more information and interactive with simulator good rather than who don't study computer engineering or computer science, or he don't have background of cryptography algorithms.

We can conclude that the project applies knowledge domain and give good results, also applies attitude domain for specific range and the results not bad, but the project suffers from weakness in apply skills domain, and the results was bad.

6

Chapter Six

Calculation and Recommendations

6.1 Introduction

The system of computer security algorithms simulator has achieved the main objectives prepared to. It is now ready to be used by the students and the interested people, the main features that achieved are:

- The system Identifies some of the security tools and algorithms.
- Using multimedia programs in learning, to show and understand how the security algorithms work.
- Designing interactive environment for cryptographic tools and algorithms, to use with face to face learning or self' reading material.
- Making the tool inexpensive or freely available.

6.2 System Achievements

Nearly all the goals of our system have been achieved. In this point the main achievements of the system are discussed and the ways of achieving it.

The project aims to build learning environment for some of cryptography algorithms, by applying bloom's taxonomy of learning, and using multimedia programs.

To achieve this goals, we build the project that simulate the most common algorithms in cryptography , we try to apply the three domains of bloom's taxonomy.

To apply the knowledge domain, the simulator contains information about every algorithm, so the user of system can identify this algorithm, how them work, compare and evaluate these algorithm.

To apply the attitude domain, the simulator contains three algorithms and every one represents type of cryptography algorithm, so the user has the ability to choose any of them, and differentiate between them.

To apply the skill domain, the user interactive with the simulator by responding with these algorithms, and choose the data that he like to encryption.

6.3 Real Learning Outcomes

After the implementation of the project we have an expert in the following points:

- Learning how to build program that simulate some of cryptography algorithms.
- Learning how to use and program in visual studio 2010. NET, and program in C# and xaml languages, and Silverlight 4.
- Learning how to apply bloom's taxonomy in learning.
- Learning how using multimedia program in learning.

6.4 Recommendations

It is recommended to add the following ideas and features on this project:

- Add new cryptography algorithms to the program.
- This project make software simulator, but another projects can make hardware simulator by using PIC microcontroller.

References :

[1] http://www.nwlink.com/~donclark/hrd/bloom.html.

[2] http://www.learningandteachinginfo/learning/bloomtax.htm

[3] William, Stallings, "Computer Security: Principles and Practice", NewYourk,2008

[4] Conklin, Arthur, "Principles of Computer Security: CompTIASecurity",2nd edition, NewYourk,2010.

[5] http://csrc.nistgov/publications/nistpubs/800_12/handbook.pdf

[6] http://hrsbstaff.ednet.ns.ca/engramja/gradcourse/multimedia.html.

[7] http://encyclopedia.jrank.org/articles/page/multimedia-in-education.html.

[8] http://portal.acm.org/ft_gateway.cfm?id=1167432&type=pdf&cfid

[9] http://encyclopedia.jrank.org/articles/page/multimedia-in-education.html.

[10] http://portal.acm.org/ft_gateway.cfm?id=1167432&type=pdf&cfid