International Journal of Sciences:
Basic and Applied Research
(IJSBAR)

ISSN 2307-4531
(Print & Online)

http://gssrr.org/index.php?journal=JournalOfBasicAndApplied

Implementation of Lossless Preprocessing Technique for Student Record System

Elmer C. Matel[a*], Dr. Ariel Sison[b], Dr. Bartolome Tanguilig III[c]

[a]*Technological Institute of the Philippines, 938 Aurora Blvd., Cubao, Quezon City, Philippines*
[b]*Technological Institute of the Philippines, 938 Aurora Blvd., Cubao, Quezon City, Philippines*
[c]*Technological Institute of the Philippines, 938 Aurora Blvd., Cubao, Quezon City, Philippines*
[a]*el_matel@yahoo.com*
[b]*ariel.sison@eac.edu.ph*
[c]*bttanguilig_3@yahoo.com*

Abstract

The Implementation of Lossless Preprocessing Technique for Student Record System of the Lyceum of the Philippines University mainly aims to apply an effective data compression algorithm to efficiently store data, improve data transmission via web-based infrastructure and ensure security of records. The system is comprised of the following modules namely: Smart ID Registration Module, Client-Side Application Module and Information Kiosk Module. For enhanced security consideration, GZIPalso known as GNU ZIP algorithm is implemented to compress records before saving in the central database. It is a lossless data compression utility that is based on the deflate algorithm with the format defined in Internet Engineering Task Force RFC1951: DEFLATE Compressed Data Format Specification version 1.32. This standard references the use of the LZ77 (Lempil-Ziv, 1977) compression algorithm combined with Huffman coding. On the other hand, the lossless decompressionrestores the data by bringing back the removed redundancy and produces an exact replica of the original source data. Results of the software evaluation indicate high acceptability on the overall system performance.

Keywords:compression; decompression; security, smart card/ID; authentication; identification; interactivity

----

* Corresponding author. Tel: 09175665487

E-mail address: el_matel@yahoo.com

1. Introduction

In times when progress and development of day to day business transactions are fueled by dynamic and innovative solutions, data storage and security should be given an utmost consideration. With the advent of technological advancement and major enabling opportunities that streamline business processes, comes new and exciting challenges including data preservation. Network security and database utilization play significant roles in protecting the confidentiality of the documents against unlawful use. The main objective of the network security is to avoid the misuse or unauthorized access with the intention of extracting information for personal use. No matter how fast today's computers and networks are, the users will constantly need faster and faster services without compromising its security features. To reduce the volume of data for transmission, compression algorithm is implemented. Aside from the advantage in transmission time, it also ensures data security.

The algorithm of a Lossless data compression technique guarantees full reconstruction of the original data without any distortion in the process [1]. This technique uses the concept of preserving the source data accuracy by eliminating redundancy from the source data application. The original source data is reconstructed from the compressed data. The compression process restores data by bringing back the removed redundant data and produces an exact replica of the original source data. The quantity or amount of redundancy removed from the source data varies and is highly dependent on the source data statistics, which are often non-stationary.

Aside from the concept of data compression, the researcher also applied Smart card technology [2] in authenticating the users of the system. Though its concept was introduced way before 1960s, still it has a vast array of applications that are yet to be unraveled and benefited from. It is a pocket-sized card with embedded integrated circuits used to provide identification, authentication, data storage and application processing. This add-on feature to the world of information technology can provide an ease when it comes to the portability, security and convenience of every stakeholder of an organization. Through decades of research and development, smart card technology is proven as the most secure and potent form of authentication or identification. Applications include banking facilities, accessing secured systems, vending, purchasing and other various purpose.

The Lyceum of the Philippines University (LPU) Cavite campus envisions itself as a leading university in the Asia-Pacific region, dedicated to the development of the integral individual who constantly seeks the truth and acts with fortitude in service to God and country. Part of its mission is to provide equitable access to learning through relevant, innovative, industry-based and environment-conscious programs and services in the context of nationalism and internationalism, however, problems are inevitable. (LPU)-Cavite has been a remarkable institution especially in the province of Cavite in terms of providing high quality learning and services to its students. Aside for being the *"First and Only Resort Campus in the Philippines"*, LPU finds its way in making serious studying made exciting. With this, further researches and studies must continuously be conducted to support its need along with the growing population. From the initial one thousand five hundred enrollees during the first year of operation, LPU has grown significantly and now caters more than eight thousand five hundred students and still counting in just six years. State-of-the-art facilities and IT infrastructure could be contributing factors in attracting students to enroll; however, data encryption and other form of data compression of online information have never been considered. In order to satisfy the major stakeholders, proficient and highly reliable data security system implementation is greatly needed. At present, students at LPU are using a contactless smart ID which they use to get in and out of the campus through the turnstile.

Focusing on the security aspect, students are using smart ID [3] but its full potential is not maximized. It is used only in getting in and out of the campus through the turnstiles that actually verifies the validity of the ID. One time registration of the ID is implemented as part of the enrolment procedure but due to significant increase in population, data storage capacity could be an issue in the future since no data compression technique is in place. Since LPU is moving towards globalization, online IT infrastructureis already established [4]. A lot of school transactions can be done in the comfort of one's living room; however, efficient transmission time and security measures must be implemented as soon as possible to avoid misuse of information by hackers and other unauthorized users. Another area where problems are normally observable is during seminars, workshops, trainings and other activities that require registration of participants. It takes an hour or more just to verify the attendance of an individual not to mention the delay in the release of certificate due to manual encoding of information.

In order to facilitate such innovation in terms of security and data compression, students must undergo a mandatory smart ID registration as part of the enrolment system. This is the *Registration Module* where information will be stored in the central database and shall be assigned with unique code found in every card. This unique code will be the primary key that distinguishes a certain student from the rest. Part of this module is the Faculty Registration where information of different faculty members is encoded in the system. During this period, data compression is applied. The *Client-side Application Module* offers a convenient way of capturing students' attendance in a seminar, conference, general assembly or any event that requires registration. All they have to do is to budge-in their IDs. Attendance report and certificate of participation can be generated automatically with minimal human intervention. The administrator can just simply upload the template of the certificate. Part of this module is the Faculty panel where grades of the students are encoded. Since the system will be running in a web environment, users can access their account at the *Information Kiosk Module* where authentication thru password is needed. Initially, password is automatically assigned upon registration; however, the user can change the password any time since the system is working on a web-based setup. Once logged in, students can view their grade report summary as well as the subjects to be taken in the following semesters [5].

1.1 Objectives of the Study

The general objective of this study is to apply robust security algorithm using lossless pre-processing data compression technique for Student Record System of the Lyceum of the Philippines University.

Specifically the study aims to:

- implement an effective data compression/decompression algorithm tomaximize storage capacity, improve data transmission and secure important records;
- develop an electronic means of capturing students' information during school events with minimal

human interventions using smart ID technology;

- apply quality standardin evaluating software performancein order to address some well-known human biases; and
- provide an interactive way of displaying information without compromising the confidentiality of student's record.

1.2 Significance of the Study

The 1st semester of Academic Year 2014-2015 marks the 7th year of solid commitment of LPU-Cavite in providing high quality services, multifaceted learning and character development to the students who entrusted their college development to the university. From the existing remarkable number of students poses several conflicting issues mainly in security, data management and report generation. The existing IT infrastructure of the University does not apply any data compression/decompression technique in storing and retrieving vital student information. This could leave the system vulnerable to attacks from unwanted users.

To proactively address these underlying situations, the proponent applied a security algorithm using lossless data compression for Student Record System at LPU Cavite. The idea is to compress students' record which will surely improve the security since the data are in encrypted format. Data storage capacity is significantly improved as well since data compression makes the file size smaller. This proposed system runs in a web environment to accommodate remote usage of the system within the campus such as the attendance registration process during an event and the Information Kiosk.

The system is comprised of the following modules namely: Smart ID Registration Module, Client-Side Application Module and an Information Kiosk Module. Contactless smart ID with proximity of one inch is used to trigger an event in the system. This smart card is used to interact with the system. The proposed study is designed to accommodate only three levels of access. First is for students who are only limited in accessing his/her own account as well as changing passwords. Second is for the faculty members who have the privilege to do the following: encode grades of students, upload templates of the certificate to be used during events inside the campus, generate reports and post announcements. And third is for the Admin who has the rights to add/edit faculty members, colleges, departments, course, sections and year levels.

1.3 Scope and Delimitation of the Study

Data security is the main consideration of the study. Basically, the study is focused on the compression and decompression procedures using lossless data compression technique. The study is composed of three main modules namely: Smart ID Registration module, the Client-side Application module and the Information Kiosk module. During the smart ID registration, the researcher will be implementing the use of ACR122U NFC reader which is the world's first NFC reader compliant with the CCID specification. The ACR122U and its token version, ACR122T, are designed to support not only Mifare and ISO 14443 type A and B cards, but also FeliCa and NFC tags. A unique code embedded inside the Smart ID is assigned to a specific student. Compression technique is then applied before data is stored in the database. The proposed system starts in the Smart ID registration before the opening of the academic year. Only enrolled students and active employees/staff are allowed to be registered in the system. Smart ID holder is required to give his/her own password as part of the security feature and has the option to change or edit profile. A Client-side application is used as entry point of vital information to be stored in the common database. An Information Kiosks module is also included in the proposed system where students can check the status of their records. Before the user makes a successful inquiry in the Kiosk terminal, Deflate algorithm decompresses the information and returns it to its original form. With this procedure, data running in and out of the network is secured. Other units and departments can be linked for future expansion. The proponent uses contactless smart reader and a smart card with total capacity of 752 bytes divided into 16 sectors.

The extent of the study is limited to LPU Cavite undergraduate programs only since the Graduate School is an independent entity.The system does not include any functionality that involves financial transactions such as payroll, bookstore payment systems and the likes. Image processing is covered in the proposed system since the images of the students are very minimal and already in the compressed format.

2. Design Concept

2.1 Review of Related Literature

2.1.1 Smart Card Technology

A smart card or sometimes referred to as "*smart ID*" is a card made of plastic usually polyvinyl chloride that has an embedded computer chip – either a memory or microprocessor type that actually stores and even transacts data. The transfer of data is via card reader that interacts with the system. Contactless smart card technology is a breakthrough in card security that does not require physical contact between a card and reader. This innovation offers advantages to both the organization issuing the card and the cardholder themselves. One possible benefit for the issuing organization is that the card can support multiple applications on a single card, integrating mix of technologies and supporting a variety of security policies for different situations [6].

Smart card provides identification, authentication, data storage and application processing. It may also provide strong security authentication for single sign-on (SSO) within large organizations either contact or contactless card design.

The authors in [7] made a research regarding the needs of the universities of simple identity cards for all employees and students who are granted access to certain data, equipment and departments according to their status. A multifunctional microprocessor-based smart card can be used to incorporate identity with access privileges at the King Fahd University of Petroleum and Minerals (KFUPM). The vision of the smart card program is to provide access to services that is secured, fast, friendly, easy to use, flexible, personalized and accessible to the users from any place at any given time.

In the study [8], smart card technology in the University of Mubai Library is used to monitor user behavior, library traffic, library planning, physical security and managing e-purse for cashless transaction. This paper is aimed at conducting pilot study for implementation of smart card for access control to safeguard both physical and intellectual property. Reports are compiled and reviewed for effective and efficient collection development, weeding out and library planning; however, SOUL 2.0 is the software used for computerizing the library in-house activities where the smart card technology is integrated.

In the paper presented by [9], a software solution was implemented for Student Management Information System at the University of Prishtina. The study is centered in the application of X.509 digital certificate to digitally sign the student grades as a form of security measures and store the information in a smart ID. Unlike other user authentication scheme that applies static information, the study by authors in [10], present a dynamic ID-based remote user authentication scheme using smart cards. Individual user possesses a smart card for later login and authentication. The user submits the identity and password associated with the smart card during the login phase. In the authentication phase, the system verifies the remotely encoded password to check its validity. This study allows the user to choose and change their passwords freely and do not maintain any verifier table. The scheme applied a strong secure procedure against ID-theft, and can resist the reply attacks, forgery attacks, guessing attacks, insider attacks and stolen verifier attacks. In the study on Image Compression Using Harmony Search Algorithm [10], image compression technique eliminates redundant information in an image which in effect actually minimizes the physical space requirement of the image. With the use of the Harmony Search Algorithm (HSA), a meta-heuristic optimization algorithm was applied as the underlying procedure for image compression. This was inspired by the music improvisation process of musicians. As a result, HSA is feasible to use as an algorithm in compressing images.

2.1.2 Image Compression

The authors in [11] made a study on advanced medical imaging that requires storage of large quantities of digitized clinical data using Dynamic Image Data Compression in Spatial and Temporal Domains. The proposed compression algorithm has three stages: 1) addressing temporal redundancies in the data through application of image optimal sampling, 2) addressing spatial redundancies in the data through cluster analysis, and 3) efficient coding of image data using standard still-image compression techniques. As a result, storage space for dynamic image data is able to be reduced by more than 95%, without loss in diagnostic quality. This means that the proposed theory and algorithm are expected to be very useful in medical image database management and telecommunication.

Run Length Encoding (RLE) for real-time image identification processing is conducted by [12] which actually compresses the image and allows the processing algorithm to efficiently identify the size and position of the object in the image. This technology is normally used for robotic vision system that enables a reliable identification of 11 objects in the 16.67millisecond sample time of an interlaced NTSC video image.

2.1.3 Data Compression Algorithm

It would, therefore, seem desirable to perform significant computation to reduce the number of bits transmitted [13]. This theory means that if the energy required to compress data is less than the energy required to send it, there is net energy savings which implies a longer battery life for portable computers. According to the result of the study, when applied to Unix*compress*, these optimizations improve energy efficiency by 51%.

In the study made by authors in [14] discovered a new technique for text data compression using the utilization of ternary representation of numbers to compress text data with fixed-symbol-length coding techniques. Binary map for ternary digits was used. For alphabet, the 4-digit ternary representation with lowercase and uppercase letters was implemented. As a result, the length of the bits string was minimized.

In the study entitled *"Encoding and Compression for the Devices Profile for Web Services"* made by [15], there is a big challenge when deploying further application layer network protocols on top of 6LoWPAN is the message size of existing - mostly XML based - protocols which does not meet the resource requirements of deeply embedded devices without further research efforts. This paper presents different data compression techniques for the Devices Profile of Web Services (DPWS) to be applied in 6LoWPAN networks. Therefore a realistic scenario was analyzed to determine 18 types of messages that were compressed and encoded using existing schemes and tools. For the first time, an Efficient XML Interchange (EXI) format for DPWS was investigated[15].

According to the proponents in [16], compression and differencing techniques can greatly improve storage and transmission of files and file versions. Vcdiff encoding format was used in this study for differencing and compressing data. Empirical study also shows the effectiveness of the said algorithm.

2.1.4 Security Implementation

A study on lossless compression data format focuses in the compression of data using a combination of the LZ77 algorithm and Huffman coding as a security measure [17]. As a result, data can be produced even for an arbitrarily long sequentially presented input data stream, using apriori bounded amount of storage location. The author of this study made another research entitled "GZIP file format specification version 4.3" which focuses on specification that defines a lossless compressed data format that is compatible with the widely used GZIP utility that includes a cyclic redundancy check value for detecting data corruption. This GZIP file format uses the DEFLATE method of compression but can be easily extended and can be implemented readily in a manner not covered by patents.

According to the authors in [18], an improved Data Encryption Standard (DES) can be developed by incorporating ODD/EVEN bit conversion to an existing DES algorithm. The said algorithm will provide better security scheme in protecting the data in smart cards from unauthorized access and possible modification. One distinctive characteristic of this study is that the program simulation provides a good start in robust encryption technique that does not require too much mathematical computation. To improve database performance a study was conducted by authors in [19] that applies data extraction using preprocessing compression based on Lempel-Ziv-Welch (LZW) algorithm. Results show a faster transmission rate due to its data compression effect without compromising the integrity and reliability of data.

A paper entitled *"A simple lossless preprocessing algorithm for hardware implementation of Deflate data* *compression"* [20] was intended to explore possibilities of developing efficient use of storage space and real time data compression solution. Using Deflate preprocessing algorithm, the researchers were able to improve the compression ratio by 3.39% making it suitable for hardware implementation.

2.1.5 Lossless Compression Algorithm

A lossless data embedding method can be used as a new paradigm in digital watermarking [21]. The researchers were able to conclude that the current data embedding method could cause inevitable distortion to an original image. This distortion cannot be removed completely due to quantization, bit-replacement and even the truncation of grayscales ranging from 0 to 255. Part of their findings reveals that the distorted image may not be used for medical imagery or for military image inspected under nonstandard viewing condition. To address this problem, the researchers applied lossless data embedding method that will completely remove watermarked image after embedded data has been removed.

Compression can reduce the number of bits that need to be transferred and stored. However, the runtime overhead due to compression may be undesirable in high-performance settings where short communication latencies and high bandwidths are essential. This paper entitled *"Fast Lossless Compression of Scientific Floating-Point Data"* describes and evaluates a new compression algorithm that is tailored to such environments. It typically compresses numeric floating-point values better and faster than other algorithms do [22].

2.2 Conceptual Framework

The conceptual framework of the study explains the concepts, assumptions and theories that support the overall expectations of the system. It is used to visually explain the overall concepts of the system to the user. The primary aim of the study is to apply the Lossless pre-processing algorithm in compressing and decompressing of students' record to maximize storage capacity, better data transmission via web-based infrastructure and more secured records. The proponent used the Input Process Output (IPO) chart as illustrated in Figure 1 below. Starting from the Input box where the Knowledge requirements is included, this contains the initial steps that are essential for the next phases, compression/decompression concepts and applications, technologies used, database management. Next is the Software requirements that contains software specifications for web-based and windows application modules. And lastly for this area is the Hardware requirements which enumerates the technical specifications of the hardware components of the system.The Process box contains the necessary procedures to be undertaken by the system. This involves the incremental method requirements that contain the requirements analysis needed as well as the modifications made by the proponent, design and development which depict the general to specific flow of the system, tests applied and implementation procedures. And finally, the Output box indicates the expected outcome of the project which is the fully functional application of the implementation of the lossless algorithm.

Figure 1 shows the overview of the system flow of the study "*Implementation of the Lossless Pre-processing Technique for Student Record System*" which depicts the relationship of the specifications needed, procedures and processes involved, technologies adopted, testing procedures conducted and the expected results.

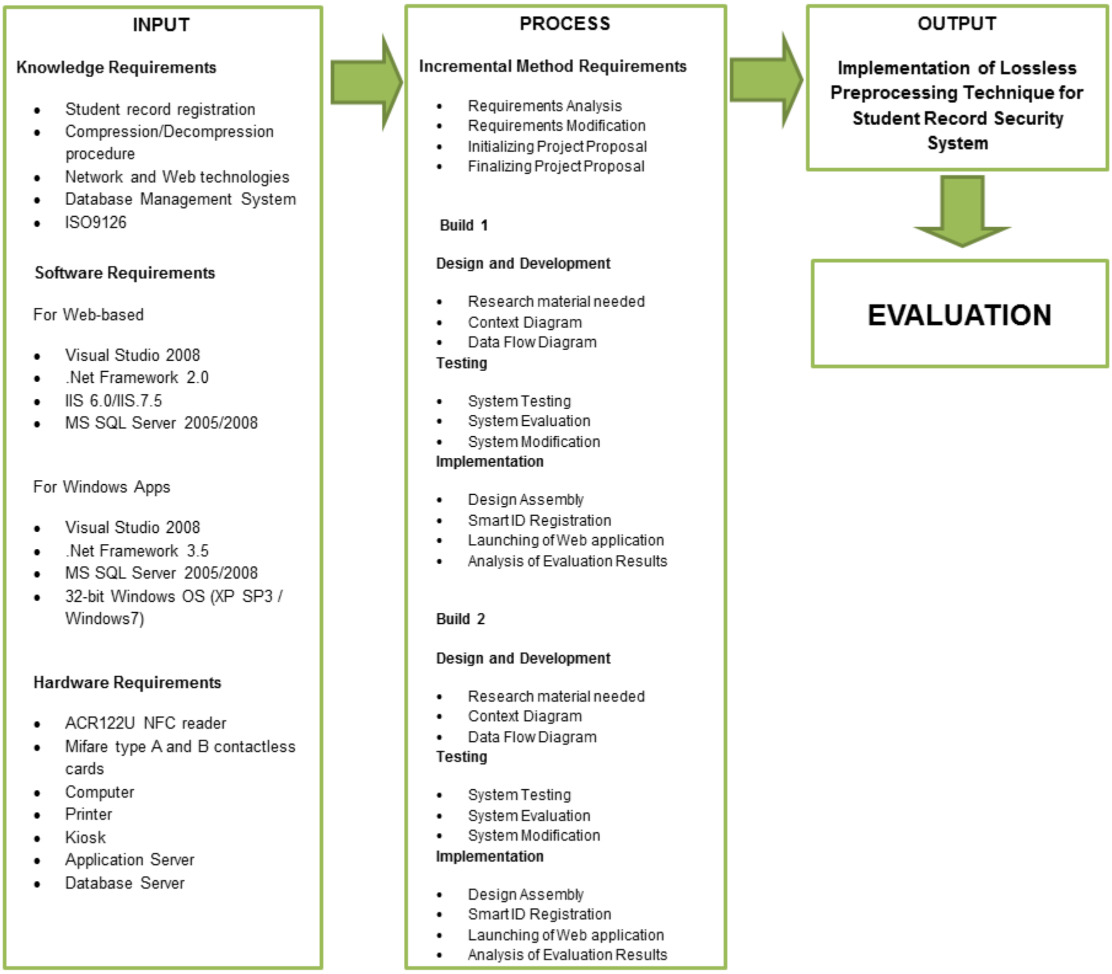


Figure 1. Input Process Output (IPO Chart)

2.3 Definition of Terms

*AUTHENTICATION***-** It is the process of confirming the validity and truth of an attribute of an entity that often involves verification the validity of identification of the LPU Cavite students. In private and public computer networks, authentication is commonly done through the use of log in passwords. Knowledge of the password is assumed to guarantee that the user is authentic.

Each user registers initially, using an assigned or self-declared password. On each subsequent use, the user must know and use the previously declared password. The weakness in this system for transactions that are significant (such as the exchange of money) is that passwords can often be stolen, accidentally revealed, or forgotten [23].

COMPRESSION– This is the reduction in the size of selected student data or the entire transmission unit in order to save storage location and faster transmission time. This can be done by just simply removing all extra space characters, inserting a single repeat character to indicate the presence of repeated character and substituting smaller bit strings of frequently occurring characters as in the case of GZIP algorithm.

CONTACTLESS SMART CARD– These are cards with embedded microchip that do not require physical contact between card and reader to acquire vital information. This technology is becoming increasingly popular for payment and ticketing applications such as mass transit like the LRT and MRT and motorway tolls. Most contactless fare collection implementations are custom and incompatible, though the MIFARE Standard card from NXP Semiconductors has a considerable market share.

DECOMPRESSION– It is the process of restoring the compressed student's record in its original form without losing any digital information.

DEFLATE– It is a data compression algorithm that uses a combination of the LZ77 algorithm and Huffman coding. Deflate is widely thought to be applicable in cases not covered by patents and has led to its widespread use of GZIP compressed files, PNG image files and the ZIP file format [24].

GZIP– Also known as GNU ZIP is a lossless data compression utility is based on the deflate algorithm is based on the "deflate" algorithm with the format defined in Internet Engineering Task Force RFC1951: DEFLATE Compressed Data Format Specification version 1.32. The standard references the use of the LZ77 (Lempil-Ziv, 1977) compression algorithm combined with Huffman encoding [25].

KIOSK– It is a small physical structure normally installed near the entrance of an establishment. Its purpose is to displays information for people who are passing by. Its etymology can be traced back from Persian which means an "outdoor pavilion".

LOSSLESS- Lossless data compression is a type of compression that allows a stream of data bits to be compressed and then later uncompressed such that the original data stream is fully recovered. Lossless data compression is frequently used in data networking and storage applications where exact reconstruction of the original data is critical to the application [25].

PREPROCESSING– the term is used to describe a preliminary processing of student's data in order to prepare it for the actual processing or for further analysis. The term can be applied to any first or preparatory processing stage when there are several steps required to prepare data for the user [26].

SECURITY- Protection against illegal or wrongful intrusion that attempts to copy or edit important records of student. The term intrusion concerns mostly with gaining access to user and company data.

3. Operational Framework

The proponent used the descriptive research to obtain information concerning the current status of the study to describe the facts with respect to variables or conditions in a situation. This is the part of the study that explains the type of research methodology used by the proponent; the data gathering used and the evaluation technique that was applied. The methods involved range from the survey which describes the status quo, the correlation study which investigates the relationship between variables, and the developmental studies which seek to determine changes over time. In addition, this part of the study discusses the business and technical models and the plans and strategies used in planning, designing, and development of the actual system.

3.1 Design of the System/Software

The technical view of the project development is discussed in this part where processes are carefully analyzed and illustrated for the research concept to conform with the design of the system.

3.1.1 Functional Decomposition Diagram

The Functional Decomposition Diagram (FDD) shows the hierarchical organization of business functions that comprise the business operations. FDD is used in determining the functional requirements of a solution that is broken-down into smaller and more manageable detailed pieces of information to further explain the functionality.

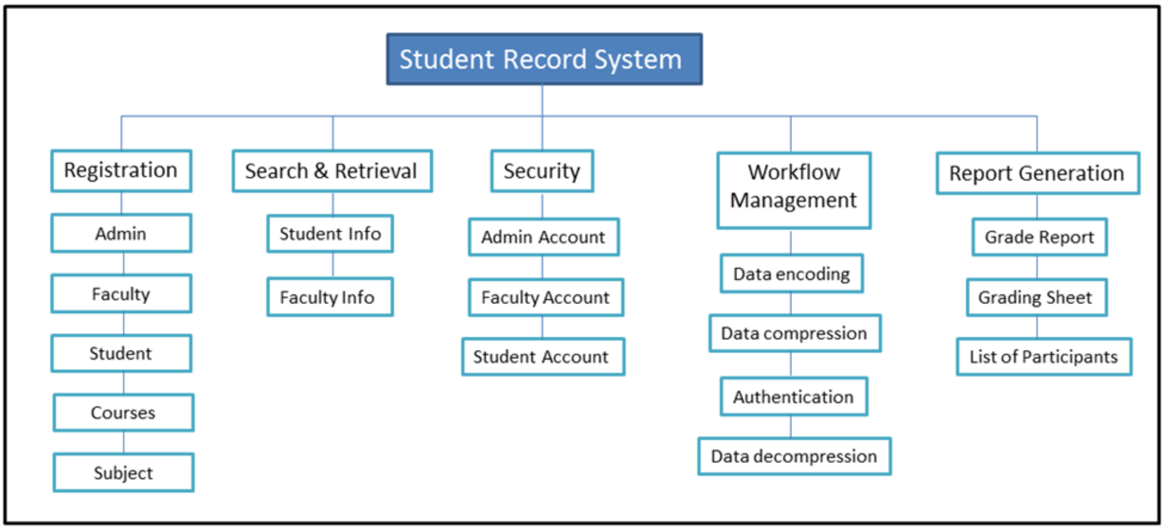


Figure 2. Functional Decomposition Diagram

Figure 2 illustrates the hierarchical arrangement of the major processes and sub-processes of the system. It shows the capabilities and functionalities of the system that are relevant the consideration of the system architecture.

3.1.1.1 Registration

This module facilitates innovation in terms of security and data compression where students must undergo a mandatory smart ID registration as part of the enrolment system. In this module information will be stored in the central database and shall be assigned with unique code found in every card. This unique code will be the primary key that distinguishes a certain student from the rest. Faculty Registration is also included in this module where information of different faculty members is encoded and compressed in the system.

a.Admin registration

The system has to verify the highest level of access to the system which is the Admin account. Adding/editing and deleting of faculty members, students, courses, subjects and curricula are the major responsibilities of the Administrator.

b. Faculty registration

In this module, the faculty members can manage their own account by initially encoding all the necessary information.

c. Student registration

This is one of the most important areas of the study. The student registration includes the basic information of the students as well as individual photos. After capturing the information, it is then preprocessed using the lossless algorithm and stored in the database. Since the information is in a compressed format, storage area can be maximized, transmission is fast and data is secured.

3.1.1.2 *Search and Retrieval*

An efficient record system can be measured in terms of easy and effective way of searching and retrieving information in the fastest way possible. This is the heart and soul of any system available. Using the contactless smart ID of the students the information is displayed instantly through the Kiosk that is installed within the campus. The unique character found inside the smart card is authenticated and verified by the system. Once verified, the compressed format of the data is reverted to its original state. Lossless algorithm has a way of restoring the compressed data without losing any important details. The same principle is applied for the Faculty records especially the encoded grades of the student.

a. Security

The main focus of the proponent is to test the actual implementation of the lossless preprocessing technique to compress and decompress students' record of the Lyceum of the Philippines University – Cavite campus. A portion of the student record was considered just to test the effectiveness of the given algorithm in terms of data integrity after compression and decompression process. Aside from this algorithm, the proponent also considered three (3) levels of access namely, the admin account, faculty account and the student account.

3.1.1.3 *Workflow Management*

This module illustrates the technical flow on how an ordinary data is transformed in a non-readable form. The application of lossless data compression/decompression algorithm to the student records of LPU Cavite campus is the main focus to maximize storage capacity since compression would mean smaller file size, better transmission speed and more secured files.

a. Data encoding

Initially, all records of the students have to be encoded in the system including their personal information, school related data, contact details and even their photos. Each record of the student has to be assigned with a unique code found in individual chip embedded in their school ID which makes it a smart card/ID. Students can use this to access their account in the Kiosk inside the campus. Faculty individual accounts are established in this module which includes their personal information, contact details, subjects and sections handled and the grades of the students.

b. Data Compression

Once encoded in the system, the data compression process starts. The deflate algorithm's compression efficiency is a variable based on the uncompressed data stream using text-based files show compression ratios in the **2.5:1 to 3.5:1** range. Compression of this range indicates operational advantages in both storage and data networking[27].The deflation algorithm used by gzip (also zip and zlib) is a variation of LZ77. It finds duplicated strings in the input data. The second occurrence of a string is replaced by a pointer to the previous string, in the form of a pair (distance, length). Distances are limited to 32K bytes, and lengths are limited to 258 bytes. When a string does not occur anywhere in the previous 32K bytes, it is emitted as a sequence of literal bytes.

3.1.1.3.c. Authentication

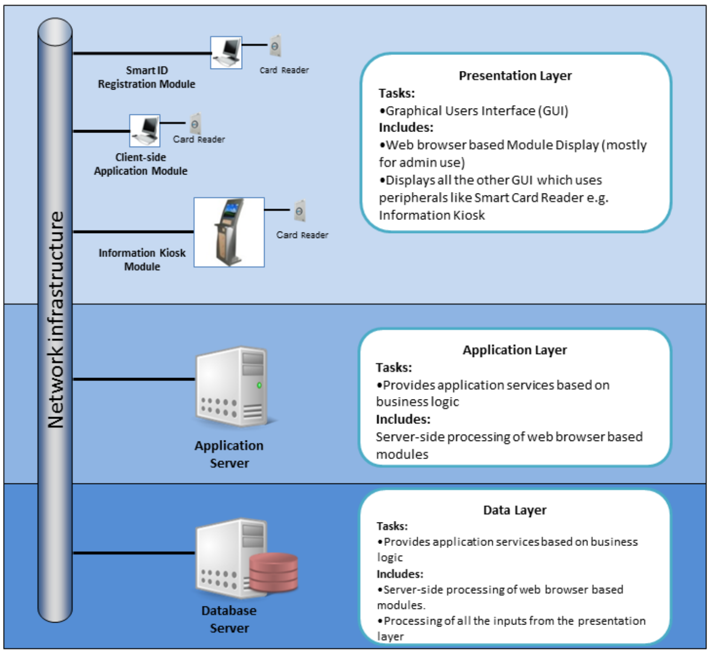
This is the most crucial part of the system where legitimate users only are allowed to use the system. Officially enrolled students will just budge-in their contactless smart ID in the card reader. The unique code in the card will be matched in the database. Once there is a match the process of decompression takes place.

3.1.1.3.d. Data Decompression

The decompression or inflate() is as crucial as the authentication procedure because it attempts to recreate the original documents. It sets up a first level table that covers some number of bits of input less than the length of longest code. It captures those bits from the stream, and looks it up on the table. The table will tell if the next code is that many bits or less and how many, and if it is, it will tell the value, else it will point to the next level table for which inflate() grabs more bits and tries to decode a longer code [28].

3.2 System Architecture

As illustrated, Figure 6 depicts the System Architecture of the proposed system. In the 1st Tier or the Presentation Layer, the Graphical User Interface (GUI) can be viewed by the users of the system. This includes the display part of the web browser-based modules such as the Administrator’s page, Faculty Grades Encoder and the Smart Card Registration Module which uses the Smart Card Reader.The 2ndTier or the Application Layer is where the server-side processing of browser-based modules is being conducted. This links the interfaces into the data layer for saving or retrieving of data from the database. The Lossless Data Compression will be implemented in this area. It is a very simple form of data compression in which the same data value that occurs in many consecutive data elements are stored as a single data value and count, rather than as the original run. This is most useful on data that contains many such runs: for example, simple graphic images such as icons, line drawings and text. And lastly, on the 3rd Tier or the data layer is where the database server for the system is being handled. In the proposed system design, there will be a centralized and consolidated database wherein all of the data from the different modules coming from different units are being stored and retrieved.



Presentation Layer
Tasks:
•Graphical Users Interface (GUI)
Includes:
•Web browser based Module Display (mostly for admin use)
•Displays all the other GUI which uses peripherals like Smart Card Reader e.g. Information Kiosk

Application Layer
Tasks:
•Provides application services based on business logic
Includes:
Server-side processing of web browser based modules

Data Layer
Tasks:
•Provides application services based on business logic
Includes:
•Server-side processing of web browser based modules.
•Processing of all the inputs from the presentation layer

Figure 6. System Architecture of the proposed system

3.3.1 Smart ID Registration Module

The full implementation of Smart ID starts with the Registration Module where the card holder has to register at the ITD as part of the enrolment procedure. Initially the card has a pre-programmed unique code to be assigned in a student number. Lossless algorithm will compress all the information. In this way, compression reduces bits by identifying and eliminating statistical redundancy. No information is lost in lossless compression. User’s profile with picture will be the main output of this process. This will be the core of the other features of the system.

3.3.2 Client-Side Application Module

This module is the data entry point of students’ grades. Since faculty members will serve as client of the system, further authentication is imposed using password. Once verified and authenticated, the user will encode all the needed data. Compression takes place once the information is submitted to the system. Faculty members can also create a template of certificate of attendance of school activities which will be used by the students using the automatic registration system. On the other side, students are also benefiting in the system since they themselves are clients. The system can be used in monitoring attendance in school events and other gatherings inside the campus that requires registration of the participants. Here in this module, attendees do not need to log in the usual way. All they need is to budge-in their smart IDs [29]. Information of the participants is automatically captured and be used to automatically fill-in the needed data in the certificate. With this added feature of the system, human error in encoding data in the certificate can be eliminated, reports can be generated instantly, registration of participants is more efficient, monitoring of attendance is effective and printing of certificate of attendance is just a click away.

3.3.3 Information Kiosk Subsystem

An Information Kiosk can be very handy for all students in monitoring their own records. For additional security, the proponent also included a password or a personal identification number (PIN) after the Smart ID is badged-in to the card reader. The Lossless Data Compression is also implemented before displaying the information from the database. Any form of data interception would not be a major threat in the system.

3.4 Research Method Used

3.4.1 Data modeling

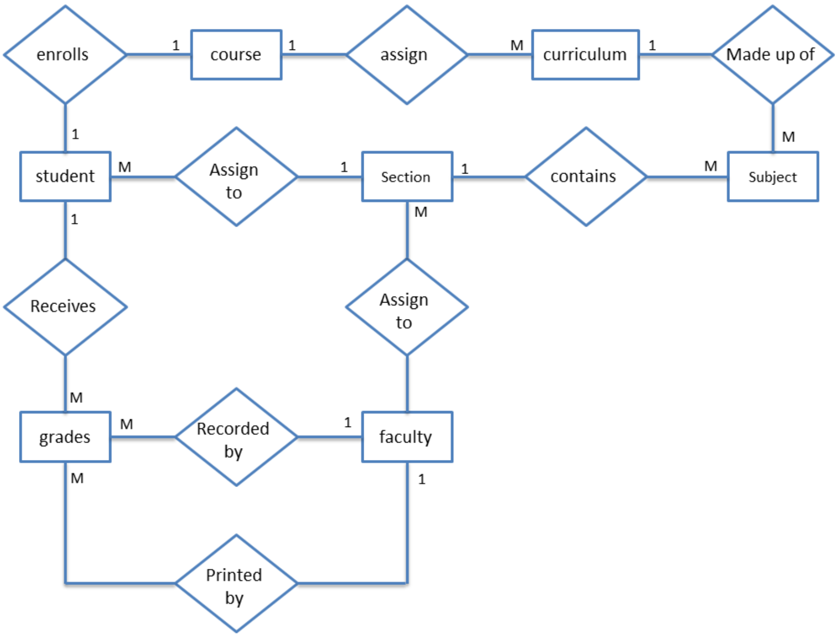


Figure 7. Entity Relation Diagram

The figure above depicts the interactions between external actors and the system to attain particular goals. Use cases are modeled using unified modeling language (UML) which is software modeling technique that defines the features to be implemented and the resolution of any errors that may be encountered.

3.4.2 Data Compression/Decompression Implementation

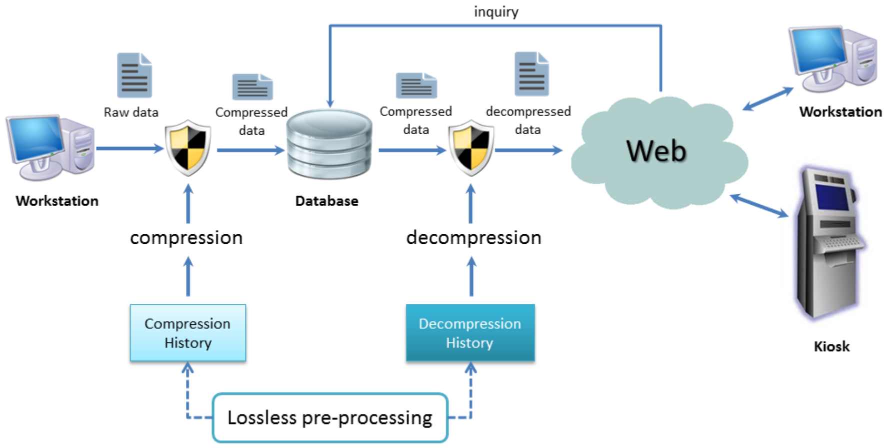


Figure 8. Data compression and decompression system flow

The conceptual overview of the system, as illustrated in Figure 8, depicts the implementation of the lossless preprocessing algorithm to compress and decompress students record as a security measure. Workstation is used to register individual smart ID to capture raw data that includes texts and images of the students. Compression is applied upon data entry to ensure security, better transmission time and smaller data size to further maximize database storage capacity. Compression history is captured to further document the internal operations [19][30]. The system also provides an interactive kiosk intended to display announcements and other information for students, however, only legitimate user can log-in to the kiosk since the system is protected by smart ID and password verification.

3.4.2.1 Lossless data compression concept

Lossless data compression is the opposite of lossy data compression. In exchange for better compression rates, lossy data compression technique allows constructing an approximation of the original data. Examples of a lossless data compression application are text documents, executable programs and source code. Image file formats, like GIF or PNG which combines the LZ77-based Deflate algorithm with a selection of domain-specific prediction filters, use only lossless compression, while others like TIFF and MNG may use either lossless or lossy methods. By principle, no compression algorithm can efficiently compress all possible data. Because of this, different algorithms came to existence that was designed with the assumptions of resolving redundancy of the uncompressed data. This includes the Deflate data compression algorithm that uses a combination of the LZ77 algorithm and Huffman coding. This has led to its widespread use, for example, in GZIP compressed files in which the proponent decided to use, PNG image files and the ZIP file format [31].

Figure 9 shows the Deflate data compression block diagram that actually breaks down the data into blocks, which may use a certain form of coding. Deflate has three options to encode original data, 1) the original uncompressed data will be taken over unchanged. The block will be introduced by header information specifying the size of the block; 2) the original byte symbols (0 – 255) are directly encoded according to the

Huffman tree that provides data codes and; 3) the sequence is addressed by the pair of parameters “sequence length” and “distance” that are both encoded using the Huffman trees in order to produce length and distance codes. Deflate data compression technique combines two algorithms together; the LZ77 and the Huffman code. The dictionary based algorithm equivalent to LZ77 is used to encode recurring sequences while the Huffman code on the other side is used for entropy coding. The procedure was introduced together with the ZIP data format such as 7z (7-Zip), GZIP and ZIP [31].

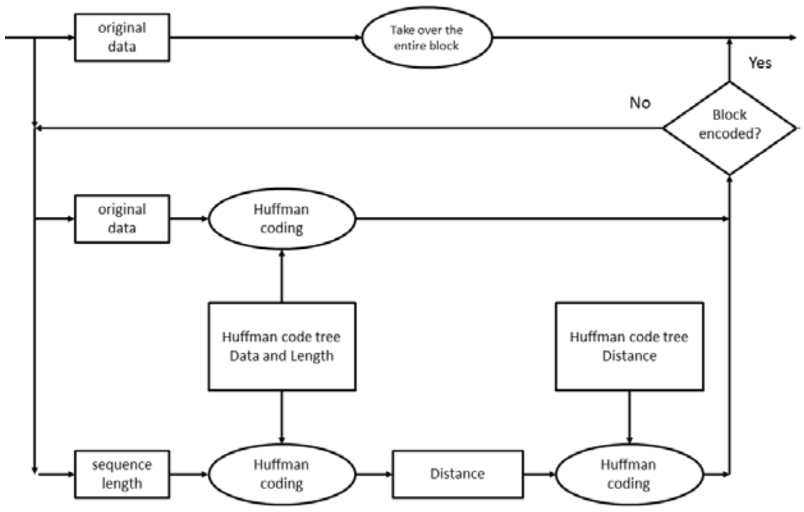
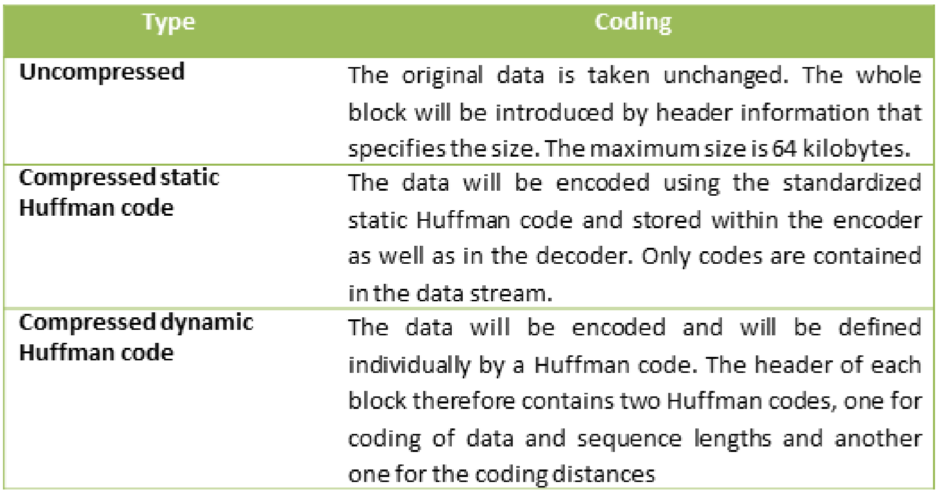


Figure 9. Deflate data compression [31]

Table 1. Deflate data structure [31]



Type	Coding
Uncompressed	The original data is taken unchanged. The whole block will be introduced by header information that specifies the size. The maximum size is 64 kilobytes.
Compressed static Huffman code	The data will be encoded using the standardized static Huffman code and stored within the encoder as well as in the decoder. Only codes are contained in the data stream.
Compressed dynamic Huffman code	The data will be encoded and will be defined individually by a Huffman code. The header of each block therefore contains two Huffman codes, one for coding of data and sequence lengths and another one for the coding distances

The table above shows that Deflate subdivides the data into blocks, which may use a certain form of coding and this has to remain unchanged within the block.

The abbreviation GZIP stands for “Gnu ZIP”. GNU is the recursive acronym for the open source operating system. The format was intended to compress single files or data streams. GZIP is designed to make the compression simple and efficient, however, creating large archives for handling wide range of file attributes is not covered. The main advantages of the GZIP versus other compression technique are much better compression and freedom from patented algorithms [17]. The GZIP utility like many other utilities such as PKZIP, WINZIP and PNG are based on the “deflate” algorithm. The standard references the use of the LZ77 compression algorithm combined with Huffman encoding. GZIP actually reduces the size of a file using the LZ77 coding algorithm.

3.4.2.2GZIP Compression Algorithm (deflate)

The GZIP looks for duplicated strings in the input data. A pointer is used to identify the occurrence of a duplicated string and form a pair representing distance and length. The distances are limited to 32 kb while the lengths are limited to 258 bytes only. If a string does not occur within the previous 32kb, it is considered as a sequence of literal bytes and is not restricted to printable characters. Literals are compressed with one Huffman tree and the distances are compressed with another tree. These trees are stored at the start of each block which can be terminated when deflate( ) method determines that it will be useful to start another block with a new tree.

When duplicated strings are found using a hash table, all input strings of length 3 are inserted in the hash table where a hash index is computer for the next 3 bytes. The longest match is identified when a hash chain for this index is not empty and all strings in the chain are compared with the current input string. To take advantage of the Huffman encoding, the hash chains are searched starting from the recent strings. The hash chains are singly linked. After a match of length *N* has been found, deflate( ) searches for a longer match at the next input byte. Once a longer match is found, the previous match is truncated to a length of one. This would produce a single literal byte and the process begins again. The match evaluation is also subject to a runtime parameter. If the current match is long enough, deflate() reduces the search for a longer match, thus speeding up the whole process. If compression ratio is more important than speed, deflate() attempts a complete second search even if the first match is already long enough [28].

When the match is not too long or when no match was found, new strings are inserted in the hash table. This is considered as fast compression that actually saves time due to fewer insertions and fewer searches.

3.4.2.3GZIP Decompression Algorithm (inflate)

The method inflate( ) prepares the first level table that contains the bits of input less than the length of the longest code. It captures those bits, compares them in the table and determine if the bits are less than or greater than the length in the table. If it is, it will tell the value, and points to the next level of that table and grabs more bits to decode a longer code.

3.4.3 Compression Simulation

Step 1:Apply LZ77 Algorithm

LZ77 compression reduces redundancy in the original data by searching for repeated sequences of bytes. Once a match is detected, the sequence is stored literally, and subsequent occurrences refer back to it.In order for implementation of LZ77 compression to be practical, repeated sequences can only be searched for within a limited window; for the Deflate algorithm, the window is defined as 32KB. In this case, LZ77 works as the ‘dictionary’ of repeated sequences beingcompressed while the sliding window scans the data [28][32]. However, it can be very time-consuming duringcompression, as many comparisons must be made between the input data and the window. On the other hand thedecompression is simple and fast, as the location to copy data from is specified explicitly.

Original Text: **“SotfProvetoolS, web hosting company that provides total web hosting solution”**

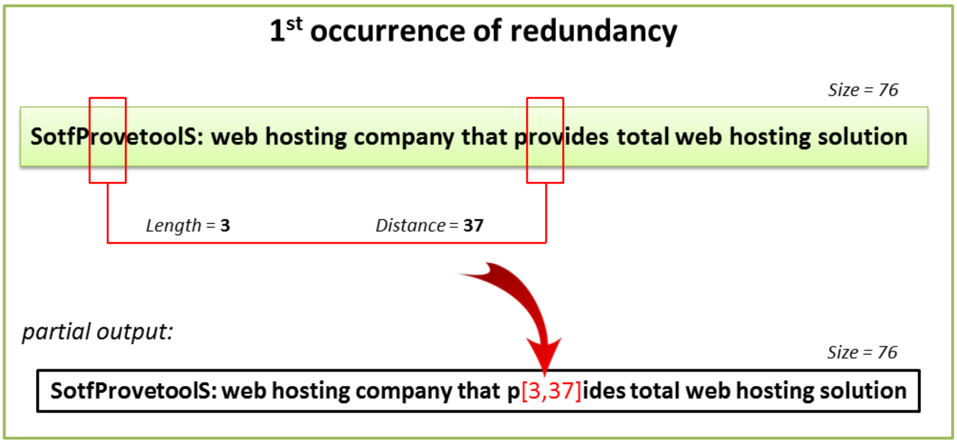

1st occurrence of redundancy
Size = 76
SotfProvetoolS: web hosting company that provides total web hosting solution
Length = 3
Distance = 37
partial output:
Size = 76
SotfProvetoolS: web hosting company that p[3,37]ides total web hosting solution

Figure 10.1st occurrence of redundancy

Figure 10 illustrates how the algorithm uses a sliding window, up to 32K bytes in length, to examine the data stream for repeating patterns. The sliding window is often referred to as history table. When a character string matches a pattern previously detected in the history table, the string is replaced with a reference to represent the backward distance of the matching string in the history table, and the length of the matching string. The matched strings are limited to a maximum of 258 bytes in length, and substitution only occurs for strings of length greater than three bytes. In this case, it is noticeable that even if the redundancy was eliminated, the size does not change because a reference is also equals to 3 bytes. When a matching string is not found, the original byte is left in the compressed data stream.

In the second occurrence of repetitive string more than 3 bytes, as illustrated in Figure 11, the sliding window or the history table captures it and extract the length of the string repetition and the distance. The *length* of the string is 5 since the spaces before and after the word also matches with the string being evaluated. The *distance* is 35.

Figure 12 also illustrates a successful substitution of reference to the repetitive word “hosting”. The *length* is just 8 because aside from the length of the word is considered; however, only the last space is accepted. The number of characters with no match is 60 and the remaining references are equal to 9 bytes, the total byte size is 69.

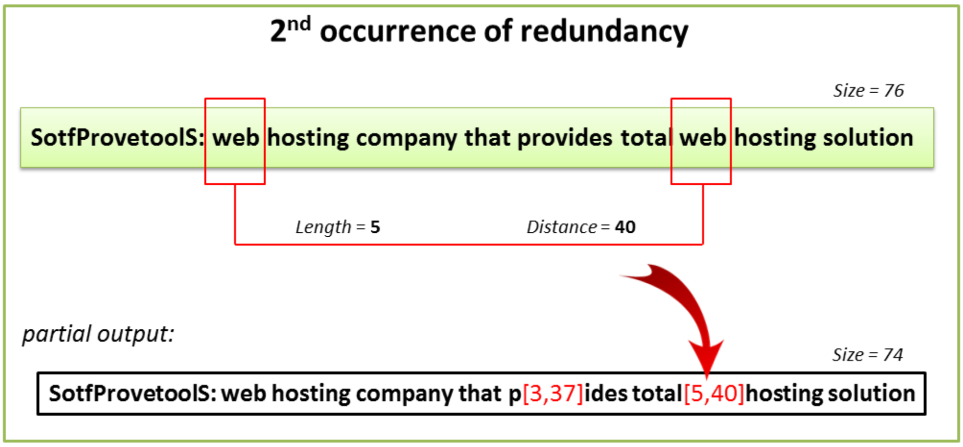

2nd occurrence of redundancy
Size = 76
SotfProvetoolS: web hosting company that provides total web hosting solution
Length = 5
Distance = 40
partial output:
Size = 74
SotfProvetoolS: web hosting company that p[3,37]ides total[5,40]hosting solution

Figure 11.2nd occurrence of redundancy

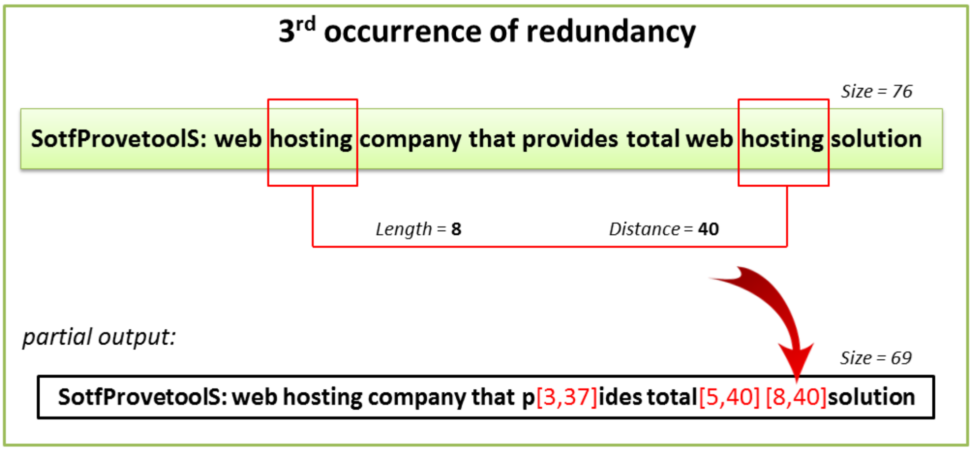

3rd occurrence of redundancy
Size = 76
SotfProvetoolS: web hosting company that provides total web hosting solution
Length = 8
Distance = 40
partial output:
Size = 69
SotfProvetoolS: web hosting company that p[3,37]ides total[5,40] [8,40]solution

Figure 12. 3rd occurrence of redundancy

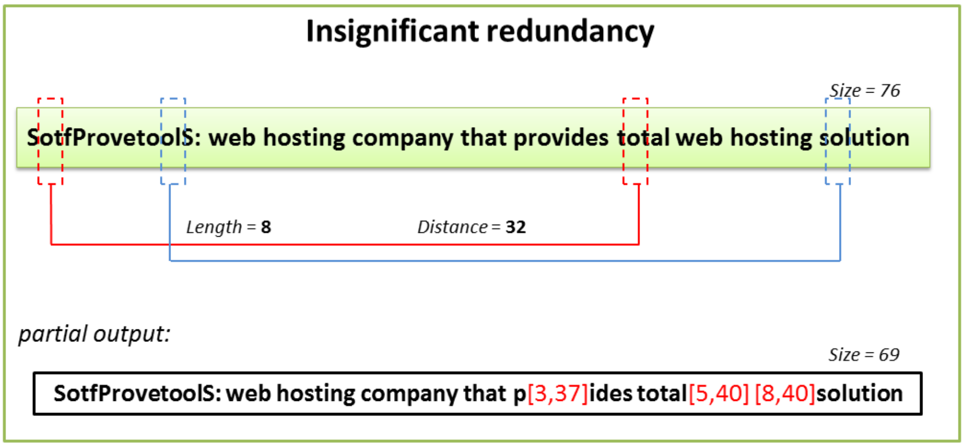

Insignificant redundancy
Size = 76
SotfProvetoolS: web hosting company that provides total web hosting solution
Length = 8
Distance = 32
partial output:
Size = 69
SotfProvetoolS: web hosting company that p[3,37]ides total[5,40] [8,40]solution

Figure 13.Insignificant redundancy

The repetition of characters with similar pattern is considered insignificant if the result of substitution would not contribute to the overall compression. As illustrated in Figure 13, manifestation of the same pattern less than or equal to 3 bytes would not have any impact in the process that is why no reference was reflected.

Step 2**:** Apply Huffman encoding

In order to reduce further the size of the compressed data produced by LZ77 algorithm, the Huffman encoding is implemented. This is achieved by using codes with fewer bits to represent the common values, and codes with more bits for the rarer values. In this case, the original string ("SoftProvetoolS") has 14 characters with a total number of 112 bits since there are 8 bits per character. The efficiency of the compression output is based on the bits reduced from the original string. The number of occurrences per character is identified as the basis for the Huffman tree implementation.

Sample string:"**SoftProvetoolS**"

No. of characters = 14

S = 2, o = 4, f = 1, t = 2, P = 1, r = 1, v = 1, e = 1, l = 1

Huffman encoding algorithm uses the output of LZ77 and group into two *alphabets*, one containing literals and lengths, the other containing distances. The characters together with their tally are arranged from left to right according to number of occurrences.

o4 S2 t2 f1 P1 r1 v1 e1 l1

The number and distribution of the symbols within these alphabets is different, so two separate Huffman encoders are used. Each compressed block is encoded using static and dynamic Huffman trees, andwhichever delivers the best results is then used.

Apply Huffman Coding

The Huffman coding tree [28] as shown in Figure 14 requires the logical arrangement of the characters from left to right. The next branch or level of the tree is determined by considering the least number of occurrences which normally are those at the rightmost. Characters and their tally are combined until the last item. The order by which the characters are arranged at the final node indicates the order of significance of occurrences per character and the number beside it denotes the number of characters in the original string.

Figure 15 depicts the Huffman tree with binary coding system. Bit 0 is assigned at the left branch while the bit 1 is placed at the right. Through this system, the compressed binary bits can be derived per character. It is noticeable that those characters that appeared more often in the string (e.g. letters o and S) have smaller binary values compare to those who appeared less or just once (e.g. letters f, P, r, v, e and l). The binary values are collected together and compared to the original string to determine the percentage of compression.

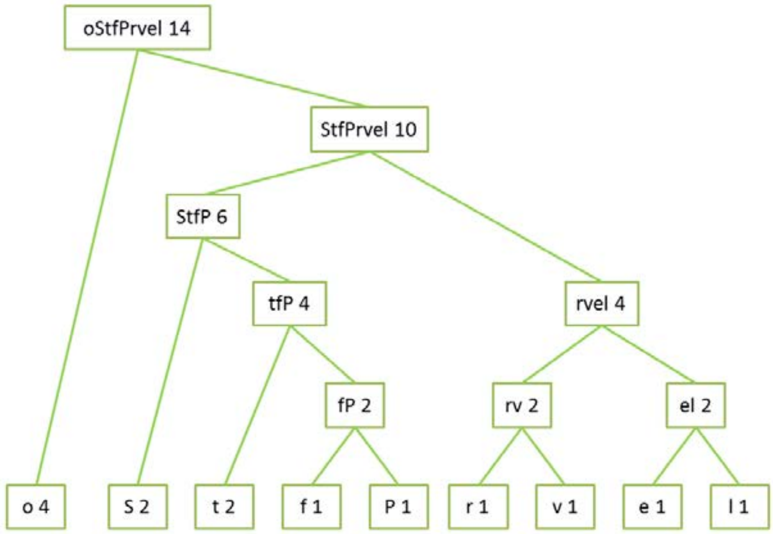


Figure 14.Huffman coding tree

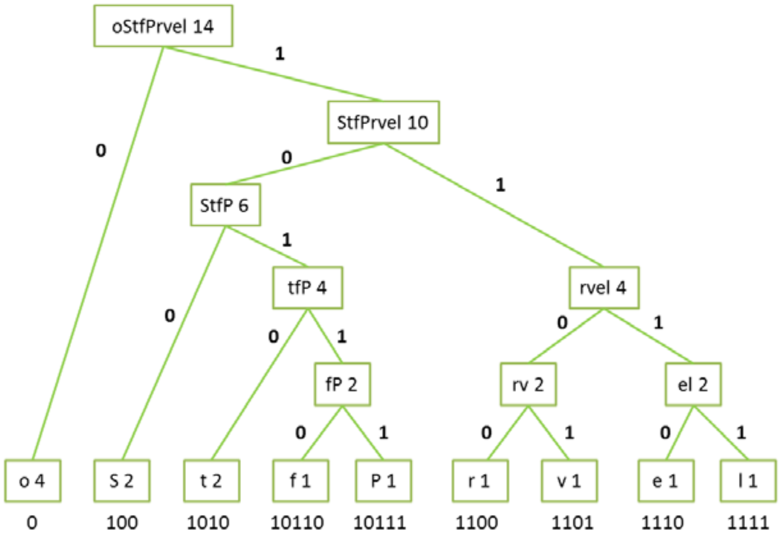


Figure 15.Huffman tree with binary coding system

Figure 16 shows the derivation of binary equivalent generated by the Huffman tree. It is evident in this figure that characters like ***o***, ***S***and ***t*** that were frequently used in the string generate smaller bit equivalent after compression while the less common letters generate more bits like in the case of ***f, P, r, v, e*** and ***l.***

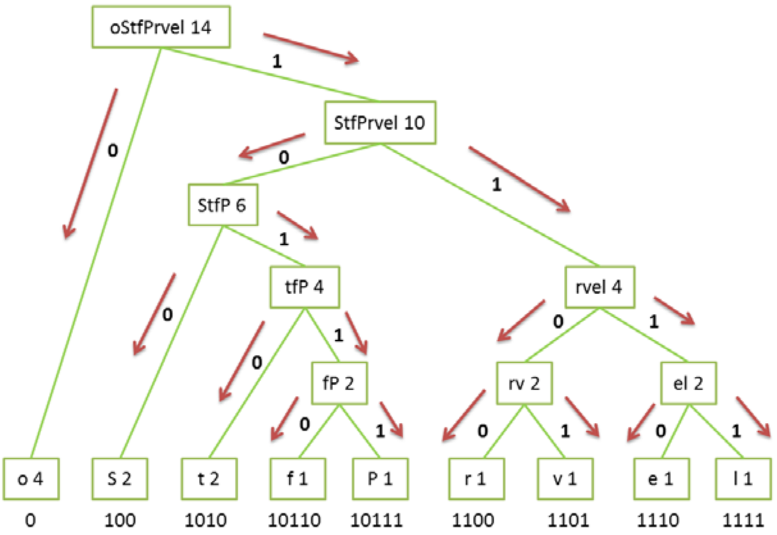


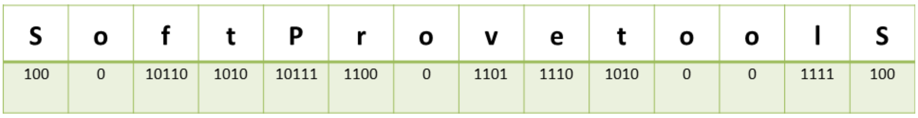
Figure 16. Huffman tree derivation of binary equivalent per code

Table 2. Huffman table

Input Levels	Input Probability	Step 1	Step 2	Step 3	Step 4	Step 5	Result
o	0.30	0.3	0.3	0.3	0.3	0	0
S	0.14	0.14	0.14	0 0.42	0	1.00	100
t	0.14	0.14	0 0.28	1	0.7	1	1010
f	0.07	0 0.14	1				10110
P	0.07	1					10111
r	0.07	0 0.14	0				1100
v	0.07	1	0.28	0.28	1		1101
e	0.07	0 0.14	1				1110
l	0.07	1					1111

The Huffman table as shown in table 2 should start by identifying the input frequency per character of the original string. This is done mathematically. Steps in combining partial results are determined based on the input frequency that normally starts from the less frequent characters.

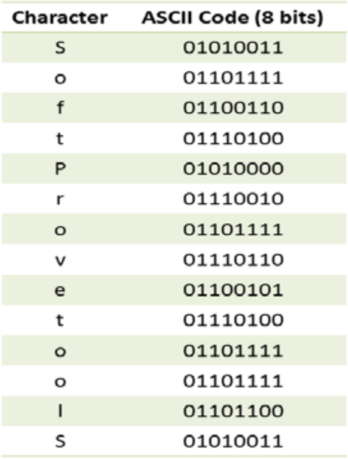
Table 3. Equivalent binary code using the Huffman tree



S	o	f	t	P	r	o	v	e	t	o	o	l	S
100	0	10110	1010	10111	1100	0	1101	1110	1010	0	0	1111	100

Table 2 shows the equivalent binary code per character derived from the Huffman tree. It is noticeable that characters with greater frequency have a shorter bit representations compared to those with lower frequencies with longer bit representations. This means that input string has characters that are more common and letters that are less common. The final step determined if the finals combination of computer frequency level per character is equal to 1.00 which is an indicator that the string being tested is completed. Just like the Huffman tree, the final output of the table is the binary code per character.

Table 4.ASCII Codification



Character	ASCII Code (8 bits)
S	01010011
o	01101111
f	01100110
t	01110100
P	01010000
r	01110010
o	01101111
v	01110110
e	01100101
t	01110100
o	01101111
o	01101111
l	01101100
S	01010011

Table 4 shows the equivalent 8-bit ASCII codification of each character in the string to be compressed. This will be the basis in comparing the original string with the compressed string to determine the effect of compression. The codification equivalent shows that the original string has 112 bits. After applying the Huffman coding, the bits were reduced to 43 bits only. That is approximately 38% reduction from the original bit size which is also equivalent to 62% savings in the storage.

SoftProvetoolS= 01010011 01101111 01100110 01110100 01010000 01110010 01101111 01110110 01100101 01110100 01101111 01101111 01101100 01010011

= **112 bits**

After compression:

SoftProvetoolS = 1000101101010101111100011011110101000111100

= **43 bits**

3.4.4 Decompression Simulation

Decoding a compressed string is much simpler and faster. The Huffman tree actually produces unique binary bits per character as it identity. The sequence of these bits as indicated in Table 2 will be the basis in decoding the original code without losing any important information.

Given: **100**010110101010101111110001101111101010001111100

S01011010101010111111000110111101010001111100

So101101010101111100011011110101000111100

Sof1010101111100011011110101000111100

Soft1011111000110111101010001111100

SoftP11000110111101010001111100

SoftPr0110111101010001111100

SoftPro110111101010001111100

SoftProv11101010001111100

SoftProve1010001111100

SoftProvet001111100

SoftProveto01111100

SoftProvetoo1111100

SoftProvetool100

SoftProvetoolS*(original input string)*

3.5 System Development Process Model Used

The use of data compression algorithm for student record security system requires series of testing and validation to determine the veracity and integrity of the output once implemented. The system design process model serves as guide on how to simplify complicated and tedious processes without compromising speed and accuracy of the output. The ideal model to be used in system development has to conform to the dynamic nature of the proposed system and the demands of the client. With this, the proponent decided to apply the prototyping paradigm.Prototyping allows rapid building of a working model of the significant portions of the system such as the data entry screens, report format and system flow. The aim is to establish an impression of the system behavior, system requirement and feasibility as requested by the client.

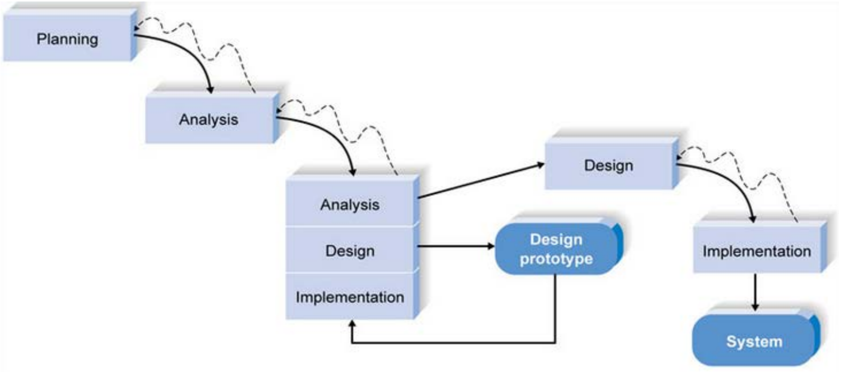


Figure 17. Prototyping Model

Information gathered in the prototyping phase allows the proponent to set priorities and redirectplans inexpensively, with a minimum of disruption. Because of this feature, prototyping and planning go hand-in-hand.

3.5.1 Planning

The Planning phase involves basic concept of the proposed system, strategies to be applied, time frame to be followed, expected reports and the implementation of security feature. Constant coordination with the client has to be done during the planning period to establish system requirements versus the expected output.

3.5.2 Analysis

One of the most crucial parts in system development is the analysis phase. A lot of things have to be taken in consideration to transform an initial hypothesis to a more solid grasp of understanding of the proposed system. Poor analysis would mean weak system performance, unreliable outcomes and definitely unsatisfied stakeholders.

In this phase, initial data gathering was conducted at the Student Record Management Department (SRMD) regarding the procedure of the system flow. The proponent also made constant communications with the Information and Communication Technology Department (ICTD) to clarify some technical issues regarding the data entry procedures.

3.5.3 Design

The design phase deals on how the system will operate, in terms of the hardware, software, and network infrastructure; the user interface, forms, and reports that will be used; and the specific programs, databases, and files that will be needed. Prototyping is a superb way to elicit feedback about the proposed system and about how readily it is fulfilling the information needs of its users. Although many aspects of the system will remain undeveloped in the prototype, the user interface must be well developed enough to enable users to pick up the system quickly.

3.5.4 Implementation

During implementation, the prototype model was built to test the solutions formulated during the planning phase. Implementation phase involves coding, testing and debugging/troubleshooting computer programs. The proponent use C# as the programming language in developing the system, however, the component where the user accesses his/her account in the Kiosk panel using the local area network (LAN) is developed using PHP scripting language. The purpose of programming is to create a program that exhibits a certain desired behavior. Coding requires expertise in many different subjects, including knowledge of the application domain, specialized algorithms and formal logic.

3.6 Testing Procedures

The testing procedure revolves in four sub-processes namely: testing plan and the actual software testing methodology in order to systematically attain the desire evaluation of the system performance.

3.6.1 Testing Plan

A Software Test Plan is a document describing the testing scope and activities. A document describing the scope, approach, resources and schedule of intended test activities. It identifies amongst others test items, features to be tested, test environment, test design techniques and entry and exit criteria to be used (http://softwaretestingfundamentals.com/test-plan/).

Part of the testing plan is the *unit tests* which deal with small units of software or modules. Like function, procedure, method. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. The Integration tests plan is focused with several units that combine into a subsystem. To make sure that the interaction of two or more components produces results that satisfy functional requirement. And finally the System tests plan deals with entire software performance. Defects found during the system testing are either fixed after doing thorough impact analysis.

3.6.2 Testing Methodology

The testing methodology used by the proponent is based on the software testing strategy designed to cover appropriate software quality standard. An adoptive quality standard instrument is used to evaluate software quality to address some well-known human biases that can adversely affect the delivery and perception of a software development project. Such qualities includes: accuracy, efficiency, reliability, security, user-friendliness, flexibility and validity.

The proponent reinforced the testing procedure using Functional Software Testing which actually aims to ensure every module is working properly. The compression and decompression procedures are also tested and validated. This guarantees that the system will respond to various cases and situations.

An integration test, on the other hand, is a different form of testing in which the interaction between two or more components is explicitly tested. This software test aims to check if the two applications that are supposedly complementing are working properly and are producing the expected outcomes. Integration tests verify that the components of your application work together in accordance with what is expected particularly in the integration of the smart card technology from its registration to the actual usage of the card.

3.6.3 Evaluation Procedures

Software evaluation is a type of assessment that aims to determine if the performance of the software fits the needs of a particular client. The idea is to look closely to the features offered by the software as a possible addition to the resources and other functionality currently used by the client. In this way, the competitive edge of the proposed system is clearly established. This part of the study focuses on the research methodology, data gathering technique and the evaluation of the data made by the respondents.

3.7 Research Method Used

The proponent used the descriptive research design which is sometimes characterized as a survey or normative approach to the study. Descriptive studies are of great value in providing facts on which professional judgments may be based. Series of interviews were conducted to know exactly the actual flow of enrollment in which the study is a part of. The grades computation has to be precise and should conform operations to avoid discrepancies with the output of the system. Staffs were also interviewed since their involvement in the proposed system is very important. To reinforce the data gathered thru interview, surveys were also administered to the students, faculty members and IT professionals. This is an excellent feedback mechanism tool to know the perception of the stakeholders towards the subject.

3.7.1 Locale of the study

The Lyceum of the Philippines University (LPU) Cavite located at General Trias, Cavite is the intended target of implementation of the system. A series of interview sessions were made with the staff and personnel in order to have a clear grasp of the details. The two departments namely: Student Records Management Department

(SRMD) and the Information and Communication Technology Department (ICTD) made significant contributions in drawing the bigger picture of the existing system as well as opportunities for improvement using new technology. In order to reinforce the data gathered from these interviews, surveys were also administered to the students and faculty members, being the direct stakeholders. For technical feedbacks and evaluations, several IT professionals with strong industry backgrounds in the field of software development participated in the survey.

3.7.2 Statistical Treatment of Data

The descriptive statistics using the mean standard deviation frequency and percentage was employed to present the demographic characteristics of the respondents and level of the system acceptability.

3.7.2.1 Mean

This is the average of the scores – the mathematical center of a distribution. It used symmetrical, unimodal distributions of interval or ratio scores. The formula for mean is:

$$X = \frac{\Sigma x}{N}$$

where:

Σx = sum of all scores

N = number of score

3.7.2.2 Standard Deviation

The standard deviation is a measure of variability used to identify whether the SD is homogenous or heterogeneous. The formula for standard deviation is:

$$\textit{Standard Deviation } (\sigma) = \sqrt{\frac{\sum(x_i - \mu)^2}{n}}$$

where:

x_i = each individual number

μ = mean

n = quantity of numbers in the group

Σ = summation (addition) sign

Table 5. Types of data and Statistical tool used per criterion

Software criteria	Types of data	Statistical tool
Accuracy	Ordinal	Mean and SD
Efficiency	Ordinal	Mean and SD
Reliability	Ordinal	Mean and SD
User-friendliness	Ordinal	Mean and SD
Security	Ordinal	Mean and SD
Flexibility	Ordinal	Mean and SD
Validity	Ordinal	Mean and SD

Table 5 shows the type of data used per criterion. The Mean and Standard Deviation (SD) will be applied as the statistical tools in deriving the overall performance of the system based on the adoptive software quality evaluation instrument.

3.7.2.3 Scale

The forty-two (42) respondents evaluated the system using the following scale:

5 – Strongly Agree

4 – Agree

3 – Slightly Agree

2 – Disagree

1 – Strongly Disagree

The obtained mean was interpreted using the following scale:

Numerical Rating	Scale Adjectival Rating
4.50 – 5.00	Strongly Agree
3.50 – 4.49	Agree

2.50 – 3.49	Slightly Agree
1.50 – 2.49	Disagree
1.00 – 1.49	Strongly Disagree

4. Results and Discussion

This chapter mainly focuses on the presentation of results, the analysis of these results and the interpretation of the overall performance of the system as seen by clients and other stakeholders.

4.1 Presentation of Results

The proponent carefully analyzed the distribution and participation of potential clients and stakeholders to the system. Out of the forty-two (42) respondents, twenty six percent (26%) are faculty members; twelve percent (12%) are from the non-teaching staff and administrators; fifty percent (50%) from students and another twelve percent (12%) from respectable IT practitioners.

4.1.1 Findings

- The implementation of lossless data compression algorithm for students' textual data and images actually improve the storage capacity of the system.

- Transmission time of the information was also improved significantly.

- The robust security algorithm ensures protection of students' record against hackers in a web-based environment.

- The integration of the smart ID/card to electronically capture students' information during school events is very useful and effective since human interaction with the system is minimized.

4.1.2 Analysis of Results

The proposed system was evaluated by forth-two (42) respondents composed of faculty members, non-teaching staffs, students and IT industry practitioners. The bulk of the evaluation was assigned to students (50%) since they are the one's to actually use the system using their smart ID/card. This is followed by the faculty members (26%). Their participation in the system is equally important especially the encoding of grades of their respective students. Administrative and non-teaching staff from the SRMD and the ICTD (12%) also contributed in the evaluation. And to determine feedbacks and evaluation of the technical side of the system, IT industry practitioners (12%) also collaborated in the study. This is to ensure a wider perspective and understanding on the underlying issues.

4.1.3 System Acceptability

The overall performance, interpretation and impression of the proposed system were attained using standard software criteria namely: Accuracy, Efficiency, Reliability, Security, Flexibility and Validity.

4.1.3.1 Accuracy

Accuracy refers to the correctness of the output generated by the system and the processes involved.

Table 6. Interpretation of Mean and SD for Accuracy

Instrument	Mean	SD	Interpretation
1. The system is capable to do correct and accurate calculations (if needed).	4.52	0.61	Strongly Agree
2. The system is error-free.	4.43	0.57	Agree
3. The system generates the correct output in all types of reports.	4.51	0.56	Strongly Agree
Average	**4.49**	**0.58**	**Agree**

As presented in Table 6, respondents strongly agreed in the accuracy of the system in terms of calculation numerical data and reports generation. This means that the users and other stakeholders of the system are guaranteed with accurate results. The system is also error-free which respondents have agreed upon. The average mean is 4.49 and standard deviation of 0.58 with an overall interpretation of *Agree*.

4.1.3.2 Efficiency

This characteristic is concerned with the system resources used when providing the required functionality.

Table 7 depicts the interpretation of the Mean and the standard deviation (SD) of efficiency. An average Mean of 4.61 and SD of 0.70 resulted to a general interpretation of *Strongly agree*. Factors contributed to this evaluation has something to do with the consistency and correctness of results delivered over short period of time, efficient input and out and the immediate display of information. The backup facility got the lowest rating since it is not directly visible to the respondents.

4.1.3.3 Reliability

It is the software characteristic that defines the capability of the system to maintain its service provision under defined conditions for defined periods of time.

Table 7. Interpretation of Mean and SD for Efficiency

Instrument	Mean	SD	Interpretation
4. The system produces consistent and correct result in fast response time.	4.67	0.64	Strongly agree
5. The system uses efficient input needed data.	4.63	0.61	Strongly agree
6. The system produces efficient output.	4.71	0.75	Strongly agree
7. The system has efficient backup facility.	4.49	0.73	Agree
8. The system immediately displays the available data for the user.	4.57	0.75	Strongly agree
Average	**4.61**	**0.70**	**Strongly agree**

Table 8. Interpretation of Mean and SD for Reliability

Instrument	Mean	SD	Interpretation
9. The system procedure was based on the official manual of procedures.	4.42	0.70	Agree
10. The system provides the actual information of the system	4.56	0.68	Strongly agree
11. The system is capable of producing backup copies of all types of data.	4.45	0.57	Agree
Average	**4.48**	**0.65**	**Agree**

The interpretation of the Mean and SD for Reliability as presented in Table 8 shows that respondents agree that the system is reliable. With 4.48 as the average Mean and 0.65 SD, the systems performance in terms of procedures and information generated is unquestionable.

4.1.3.4 Security

Securing the data that will run in an online environment is the main purpose of the study. This software quality deals with how the system response to unauthorized access.

Table 9. Interpretation of Mean and SD for Security

Instrument	Mean	SD	Interpretation
12. The system has a security password to prevent unauthorized access of data.	4.71	0.75	Strongly agree
13. The system secures important data that only administrator's account can access.	4.62	0.60	Strongly agree
14. The system manifests protection on its entire content and structure.	4.63	0.61	Strongly agree
Average	**4.65**	**0.65**	**Strongly agree**

Table 9 illustrates a strong characteristic of the system to secure itself against unauthorized access. Having an average Mean of 4.65 and SD of 0.65 is a manifestation that the system was able to perform in accordance with what is expected. Majority of the respondents strongly agree in the security features of the system since it is the main objective of the system.

4.1.3.5 User-friendliness

This means that the user finds it easy to use the system and to make it perform the desired functions.

Table 10. Interpretation of Mean and SD for User-friendliness

Instrument	Mean	SD	Interpretation
15. The over-all design of the system is suited to the knowledge level of the user.	4.73	0.65	Strongly agree
16. The system gives positive or understandable remarks.	4.45	0.57	Agree
17. The system's report and other output are easy to understand.	4.38	0.72	Agree
18. The system provides adequate feedback/ message boxes.	4.59	0.57	Strongly agree
Average	**4.54**	**0.63**	**Strongly agree**

A system with a general evaluation of *Strongly agree* in terms of user-friendliness as shown in Table 7 is a clear indicator that the users were able to use the system properly because the design suites the knowledge level of the user. The average Mean of 4.54 and SD of 0.63 can also be attributed to reports and other output that are easy to understand.

4.1.3.6 Flexibility

The software flexibility refers to the capability of the system to adopt on any changes.

Table 11. Interpretation of Mean and SD for Flexibility

Instrument	Mean	SD	Interpretation
19. The system has an option for input changes.	4.72	0.75	Strongly agree
20. The system has an option for output needed (e.g. reports)	4.63	0.61	Strongly agree
21. The system's conceptual design can be changed showing its scope, overall architecture and relationship to other systems.	4.49	0.72	Agree
Average	**4.61**	**0.69**	**Strongly agree**

As depicted in Table 11 of the interpretation of Mean and SD for flexibility, respondents strongly agreed that the system is flexible enough to adopt changes in the input and output facilities. The average Mean is 4.61 and the SD is 0.69.

4.1.3.7 Validity

Software validity refers to the lifespan or the time frame of its implementation and usage in parallel with the reliability of its output.

A general evaluation of *Agree* was given by the respondents to the system as shown in Table 12. This has something to do with the Input, Process and Output facilities of the system and how these facilities meet with the actual procedure. The average Mean of 4.47 and SD of 0.66 means that the systems validity is still high and can be able to withstand the hand of time.

4.1.4 Summary of System Acceptability

The summary of the system acceptability explains the overall performance as reflected in the computation of the Grand Mean and Standard Deviation of the system with respect to the instrument used.

Table 12. Interpretation of Mean and SD for Validity

Instrument	Mean	SD	Interpretation
22. The system Input meets the true and actual procedure of the system.	4.49	0.73	Agree
23. The system Process meets the true and actual procedure of the system.	4.27	0.60	Agree
24. The system Output meets the true and actual procedure of the system.	4.64	0.65	Strongly Agree
Average	**4.47**	**0.66**	**Agree**

Table 13. Summary of System Acceptability

Instrument	Mean	SD	Interpretation	Rank
1. Accuracy	4.49	0.58	Agree	4
2. Efficiency	4.61	0.70	Strongly agree	2
3. Reliability	4.48	0.65	Agree	5
4. Security	4.65	0.65	Strongly agree	1
5. User-friendliness	4.54	0.63	Strongly agree	3
6. Flexibility	4.61	0.69	Strongly agree	2
7. Validity	4.47	0.66	Agree	6
Grand Mean and SD	**4.55**	**0.65**	**Strongly agree**	

Table 13 clearly shows that the overall rating made by the students, faculty members, non-teaching/administrative staff and IT practitioners is *Strongly agree* with a Grand Mean of 4.55 and Grand SD of 0.65. This rating can be attributed to the system's Efficiency, Security, User-friendliness and Flexibility with an average Mean of 4.61, 4.65, 4.54 and 4.61 respectively. However, the Accuracy, Reliability and Validity got an overall rating of *Agree* which does not make the system less significant. It is also evident that Securitycriterion got the first rank in the evaluation. This is because the main object of the system is to implement robust security feature for student record using lossless data compression.

5. Summary, Conclusion and Recommendation

5.1 Summary

The Lossless data compression technique guarantees full reconstruction of the original data without any distortion in the process. This technique uses the concept of preserving the source data accuracy by eliminating redundancy from the source data application. The original source data is reconstructed from the compressed data.

The proponent used three modules for the study namely: Registration Module, Client Application Module and the Information Kiosk Module which area equally important in the system implementation. Part of the added technology is the use of smart ID/card the bears the unique access code to student's individual records.Contactless smart ID with proximity of one inch is used to trigger an event in the system. For enhanced security consideration, GZIPalgorithm compresses student's record before saving in the central database.Authentication procedure is invoked if an attempt to use the system is activated. Once the authenticated, decompression procedure comes next.

The system performance is excellent in capturing students' information during registration in school events using smart ID/card technology. Personal information and time of arrival are extracted and automatically encoded in the certificate of attendance. The result of the software evaluation using the adaptive software quality instrument indicateshigh acceptability of the system not only to students, faculty members, non-teaching/administrative staff but also to IT practitioners. Statistic shows that the overall rating is*Strongly agree*which is a clear indicator that the performance of the system is beyond the expectation, however, rooms for improvement includes better backup facility, wider coverage of the system implementation, improve the Input, Process and Output facilities and a more comprehensive presentation of the scientific calculations of the proportionality of the compressed data versus transmission time.

5.2 Conclusion

The following conclusions were drawn based on the results and findings of the study:

- The lossless data compression/decompression technique is the appropriate algorithm for the study since it restores the data by bringing back the removed redundancy and produces an exact replica of the original source data. The quantity or amount of redundancy removed from the source data varies and is highly dependent on the source data statistics, which are often non-stationary.
- A robust security feature was successfully implemented using GZIP lossless data compression utility that is based on the deflate algorithm that combines LZ77 and Huffman encoding.
- The system performance is excellent in capturing students' information during registration in school events. This is made possible using smart ID/card technology that the student has to use to budge into the

system. Personal information and time of arrival are extracted and automatically encoded in the certificate of attendance. Error in encoding of data is eliminated since human intervention is minimized.

- The result of the software evaluation indicates the high acceptability on the system performance not by the students, faculty members, non-teaching/administrative staff but also to IT practitioners.
- Based on the results and evaluation made by the respondents, the overall rating of *Strongly agree* is a clear indicator that the system performs in accordance with what is expected as was able to serve is purpose.

5.3 Recommendation

- The backup facility of the system should be given with utmost consideration since the grades of the students are very crucial in the whole operation of the institution.
- Encoding of grades online should be part of the priority plan for upgrade. At present the proposed system only allows encoding using the intranet network setup. With this feature, faculty members can conveniently encode grades at the comfort of their home. However, stronger and more sophisticated security plans has to be implemented.
- With a Mean of 4.47 the software validity got the lowest ranking. This is attributed to the Input, Process and Output facilities that have to be improved in order to prolong the lifespan of the proposed system.
- An in-depth study on the significant advantage of data compression with image processing must be incorporated. Precise scientific computation must be presented to reinforce the hypothesis in proportionality of the data size with transmission time.

References

[1] CCSDS 120.0-G-2 -2006.“CCSDS Report Concerning Lossless Data Compression”, *Green Book*, [February 12, 2012].

[2] C. Bolchini and F.A. Schreiber. “Smart Card Embedded Information Systems: A Methodology for Privacy Oriented Architectural Design”, Dipartimento di Electronica e Informazione, Politecnico di Milano, Piazza L. da Vinci, 32, 120133 Milan, Italy, December 2001.

[3] H.-C. Wu, J.-W. Chen andC.-C. Hsieh.“Creating Added Value for Smart Card Applications: The University as a Case Study”, Department of Information Management National Taiwan University, Department of Business Administration Chinese Culture University, Taipei, Taiwan, 2011.

[4] A. Yazdanpanah and M. Hashemi. (2011, May).“A simple lossless preprocessing algorithm for hardware implementation of Deflate data compression”,*Electrical Engineering (ICEE),2011 19th Iranian Conference*, vol., no., pp.1,5, 17-19. [November 5, 2012].

[5] C. Chang and T. Wu. (1991).“Remote Password Authentication with Smart Cards”,*IEEE PROCEEDINGS-E, Vol 138*, No 3, Mam, [October 16, 2013].

[6] Smart Card Alliance(2002, October), “Contactless Technology Report”,

http://www.smartcardalliance.org/%20publications-contactless-technology-report, [September 2013].

[7] T. Halawani and M. Mohandes.(2003).“Smart Card and Information Security”, Smart Card Lab and Electrical Engineering Dept. King Fahd University of Petroleum and Minerals KFUPM 1885, Dhahran, 31261, Saudi Arabia, [May 11, 2012].

[8] N.A. Raja, “Application of Smart Card Technology in the University of Mumbai Library to Monitor User Behavior, Library Traffic, Library Planning, Physical Security and Managing E-purse for Cashless Transaction”, [October 7, 2012].

[9] B. Rexha, H. Lajqi and M. Limani.“Using Smart Card and X.509 Digital Certificates for as Student Management Information System at the University of Prishtina”, Faculty of Electrical and Computer Engineering University of Prishtina, Kodra e Diellitpn., 10000 Prishtina, 2010.

[10] R.M. Daga and J.P. Yusiong. “Image Compression Using Harmony Search Algorithm”, Division of Natural Sciences and Mathematics, University of the Philippines, Visayas, Tacloban College, Tacloban City, Leyte, 6500 Philippines.

[11] D. Ho, D. Feng, and K. Chen. (1997, December). “Dynamic Image Data Compression in Spatial and Temporal Domains: Theory and Algorithm*”, IEEE Transaction on Information Technology in Biomedicine*, vol. 1, no. 4, 2014.

[12] C.H. Messom, Demedenko, K. Subramaniam and S. Gupta, “Size/Position Identification in Real-Time Image Processing using Run-Length Encoding”, USA, May 2002.

[13] K. Barr, and K. Asanovic. “Energy Aware Lossless Data Compression”, MIT Laboratory for Computer Science200 Technology Square, Cambridge, MA 02139, 2011

[14] U.N. Katugampola.(2012). “A New Technique for Text Data Compression” , is3c, pp.405-409, *2012International Symposium on Computer, Consumer and Control*, [October 7, 2013]. DOI:http://doi.ieeecomputersociety.org/10.1109/IS3C.2012.109

[15] G. Moritz, D. Timmermann and R. Stoll.“Encoding and Compression for the Devices Profile for Web services”, Rostock, Germany, 2010.

[16] D.G. Korn and K.-P. Vo. (nd). “Engineering a Differencing and Compression Data Format”, AT&T Laboratories, Florham Park, USA.

[17] L.P. Deutsch.“GZIP File Format Specification version 4.3”, Aladdin Enterprises, May 1996.

[18] A. M Sison, B.T. Tanguilig III, B.D. Gerardo, Y.-C. Byun. (2011). “An Improved Data Encryption Standard to Secure Data Using Smart Cards”, sera, pp.113-118, *2011 Ninth International Conference on Software Engineering Research, Management and Applications,* [October 14, 2013].

DOI:http://doi.ieeecomputersociety.org/10.1109/ SERA.2011.27

[19] B.T. Tanguilig III, A.M Sison, B.D. Gerardo and Y.-C. Byun. (2011). “Data Extraction Using Preprocessing Compression Based on LZW to Improve DB Performance”, sera, pp.64-67, *Ninth International Conference on Software Engineering Research, Management and Applications*, [October 14, 2013]. DOI:http://doi.ieeecomputersociety.org/10.1109/SERA.2011.28

[20] A. Yazdanpanah and M.R. Hashemi. “A simple lossless preprocessing algorithm for hardware implementation of Deflate data compression”, *Multimedia Process. Lab.*, Univ. of Tehran, Tehran, Iran, May 2011.

[21] J. Fridrich, M. Goljan, and R. Du. “Lossless Data Embedding—New Paradigm in Digital Watermarking”, SUNY Binghamton, Binghamton, NY,October 2001.

[22] P. Ratanoworabhan, J. Ke and M. Burtscher. “Fast Lossless Compression of Scientific Floating-Point Data”, Ithaca, NY, 2006.

[23] M. Rouse, “Authentication”, http://searchsecurity.techtarget.com/definition/authentication, June 2007.

[24] L.P. Deutsch. “DEFLATE Compressed Data Format Specification version 1.3”, Aladdin Enterprises, May 1996.

[25] Chipestimate, http://www.chipestimate.com/tech-talks/2010/03/23/Altior-Unzipping-the-GZIP-compression-protocol, 2010

[26] The Computer Language Company Inc., http://www.pcmag.com/encyclopedia/term/49643/preprocessing, 2009.

[27] J. Rash (2010). “Unzipping the GZIP compression protocol”, from http://www.chipestimate.com/tech-talks/2010/03/23/Altior-Unzipping-the-GZIP-compression-protocol,Octber 2013.

[28] J.-L. Gailly and M. Adler (2003), GZIP.org. Retrieve from http://www.gzip.org/algorithm.txt, October 2013.

[29] M. Fara. “On the Weaknesses and Improvements of an Efficient Password Based Remote User Authentication Scheme Using Smart Cards”, *Department of Applied Sciences and Humanities, Hindustan College of Science and Technology*, India, November 2001.

[30] R.N. Horspool and J. Corless. “Tailored Compression of Java Class Files”, Victoria, Canada, 1998.

[31] R. Seeck (2010), Binary Essence, Nienburg/Weser, Germany. http://www.binaryessence.com/dct/imp/en000240.htm.,November 2013.

[32] R. Pereira. “TimeStep Corporation, IP Payload Compression Using DEFLATE”, *The Internet Society/*, 1998.

[33] T. Bicer, J. Yin, D. Chiu, G. Agrawal and K. Schuchardt.(2013). “Integrating Online Compression to Accelerate Large-Scale Data Analytics Applications”, ipdps, pp.1205-1216,*IEEE 27th International Symposium on Parallel and Distributed Processing*, 2013.