

CHAPTER 1

Introduction

This chapter presents an overview of the project tasks in shorts including the goal and potential applications of the project.

1.1 Introduction

During the past several decades personal function has been transformed from a relatively obscure record keeping staff to central and top level management function. There are many factors that have influenced this transformation like technological advances, professionalism, and general recognition of human beings as most important resources.

A computer based management system is designed to handle all the primary information required to calculate monthly statements of customer account which include monthly statement of any month. Separate database is maintained to handle all the details required for the correct statement calculation and generation.

This project intends to introduce more user friendliness in the various activities such as record updating, maintenance, and searching. The searching of record has been made quite simple as all the details of the customer can be obtained by simply keying in the identification or account number of that customer. Similarly, record maintenance and updating can also be accomplished by using the account number with all the details being automatically generated. These details are also being promptly automatically updated in the master file thus keeping the record absolutely up-to-date.

The entire information has maintained in the database or Files and whoever wants to retrieve can't retrieve, only authorization user can retrieve the necessary information which can be easily be accessible from the file.

1.2 Background and Present State

Currently Customer Banking Management System computerized system to handle banking process with the customer. Calculations of interest and transacting items are done by computers too. But there is no same project about this topic. So, I choose my project on this

base because this world is getting digitalized and customers want to deal easily with any bank mentioning their security issue. Security is a big issue of any bank in every country. So I hope this project will help the customers to deal with the bank without hesitating about the security problem.

1.3 Objectives and Specific Aims

The objective of this interface is to manage passwords for different accounts on the internet. The user shall be able to save all the username and passwords information of the accounts he holds on the internet using this application. These details shall be saved in the database in encrypted format. This will help the user to remember different usernames and passwords for accounts on the internet. The user shall be able to add account, edit and delete account using the system. The user has to login to the system in order to use this tool. It takes user login and password. It allows user to change the password for the tool being used. This interface is developed using Visual Studio Code. The tool will allow the user to view all the different accounts he holds in a list view on the left screen. All the details related to the account like the username, password and platform shall be displayed on the right screen. The password is visible in decrypted format. The user can save any number of account information using this tool.

CHAPTER 2

Literature Review

This chapter shows the state of the art projects about different department management system in different fields.

2.1 Introduction

The literature review deals with the topics and the researches that would help to understand Password Management System the existing systems that are similar to Password Management System. The objective of this literature review is to analyze the related work to this project and mechanisms used in previous studies.

2.2 Computer-based information system

Higgins (1976) defines Computer-based information systems (CBIS) as structured systems that rely on computer hardware and software technology to collect, process, and store and distribute information. Information systems are employed to support decision making and control in an organization. Information systems can also be used to analyze problems, visualize complex subjects, and create new products. Input, processing and output are the three activities in an information system that produces the information an organization needs.

2.3 Advantages of Computer-Based Information Systems

Compute-based information systems have been in widespread use since the 1990s in industry, non-profit organizations and government agencies. These systems provide fast, centralized access to databases of personnel information, reference reading, best practices and on-the-job training, and are easily customizable to meet an organization's needs. With the Internet and technology boom of the early 21st century, use of computer-based information networks is growing faster each year. Pant, S., Hsu, C., (1995).

2.4 C language Based System

Summary form only given. With complexities of systems rising almost daily, the system community has been searching for new methodology that can handle given complexities with increased productivity and decreased times-to-market. The obvious solution that comes to mind is increasing levels of abstraction, or in other words, increasing the size of the basic building blocks. However, it is not clear what these basic blocks should be beyond the obvious processors and memories. Furthermore, if a design consists of SW and HW, the modeling language should be C since standard processors come only with C compilers. Unfortunately, the C language was developed for describing software and not hardware. It is missing basic constructs for expressing hardware concurrency and communication among components. Therefore, we need a well defined design flow, with well defined models and a modeling language that can be compiled with standard compilers and that is capable of modeling hardware and software on different levels of abstraction including cycle-level accuracy. In order to find the solution, we look first at the system gap between SW and HW designs and then try to bridge this gap by looking at different levels of abstraction. In addition, we define example semantics for each abstraction level, necessary styles of modeling and required model refinements to bring the specification to a cycle-accurate implementation level. We exemplify this by looking in detail at the RTL and SL abstraction levels. From this point of view we analyze the basic approaches in the academia and the industry today, and try to find out where we, as a design community, are going. We finish with a prediction and a roadmap to achieve the ultimate goal of increasing productivity in system design by more than 1000X and reducing expertise level needed for design of complex systems to the basic principles of design science only.

CHAPTER 3

Methodology

In this chapter the implementation procedure is stated and system environment, system design also stated.

3.1 Existing System

The password management was done manually. Users had to remember the list of passwords for different accounts on the internet. The user had to manually maintain a list of all usernames and passwords. This task was very tedious. In cases where privacy for user accounts was required, it was difficult to manually maintain this list of passwords. The user had to save a password details list in some word format so that it would help him to remember the username and passwords for different accounts on the internet.

3.2 Proposed System

This is a simple console user interface developed using visual studio code. It allows the user to add account information for different accounts he holds on the internet. It basically stores the username and password details in an encrypted format in the file system. The user has to login to the system. It checks for the validity of the user. The user shall be able to add an account. The user can add user name and password details and save to the database. The list of accounts held by the user is displayed on the left screen. When the user selects the particular account on the left panel, the details of the account gets displayed on the right screen. The password shall be displayed in encrypted format. Once the user selects the unhide option, he shall be able to view the password on screen. The user can add any number of accounts using this application. The encryption algorithm is very robust and uses the symmetric key block cipher which is very effective way of handling passwords.

3.3 Features of Proposed System

This project's features are:

- ❖ Homepage
- ❖ Creating account system
- ❖ Check user existence
- ❖ Filter user data
- ❖ Editing option of user's information
- ❖ Deleting option of user's information

- ❖ Total view of user's information
- ❖ Deleting option of user's account
- ❖ Log out option
- ❖ Exiting Option

3.4 System Environment

This project is built with the help of some hardware and software tools. They are given below:

3.4.1 Hardware Configuration

- ❖ **Operating System:** Windows 11
- ❖ **Processor:** AMD Ryzen 5 3600
- ❖ **RAM:** 8GB DDR4 3600MHz
- ❖ **Monitor:** Minimum resolution of 1024x768

3.4.2 Software Configuration

- ❖ Visual Studio Code
- ❖ GCC
- ❖ Browser

3.4.3 Code Blocks

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).

3.4.4 GCC

The GNU Compiler Collection (GCC) is a compiler system produced by the GNU Project supporting various programming languages. GCC is a key component of the GNU toolchain and the standard compiler for most projects related to GNU and Linux, including the Linux kernel. The Free Software Foundation (FSF) distributes GCC under the GNU General Public License (GNU GPL). GCC has played an important role in the growth of free software, as both a tool and an example.

When it was first released in 1987, GCC 1.0 was named the GNU C Compiler since it only handled the C programming language. It was extended to compile C++ in December of

that year. Front ends were later developed for Objective-C, Objective-C++, Fortran, Java, Ada, and Go, among others.

Version 4.5 of the OpenMP specification is now supported in the C and C++ compilers and a "much improved" implementation of the OpenACC 2.0a specification is also supported. By default, the current version supports *gnu++14*, a superset of C++14, and *gnu11*, a superset of C11, with strict standard support also available. As of GCC 9, C++17 support is no longer experimental, and it, or strictly *gnu++17*, is the default in (the upcoming) GCC 11. GCC also provides experimental support for C++20.

GCC has been ported to a wide variety of instruction set architectures, and is widely deployed as a tool in the development of both free and proprietary software. GCC is also available for many embedded systems, including ARM-based; AMCC, and Freescale Power ISA-based chips. The compiler can target a wide variety of platforms.

As well as being the official compiler of the GNU operating system, GCC has been adopted as the standard compiler by many other modern Unix-like computer operating systems, including most Linux distributions. Most BSD family operating systems also switched to GCC, although since then, some BSDs including FreeBSD and OpenBSD have since moved to the Clang compiler. macOS also switched to Clang after using GCC. Versions are also available for Microsoft Windows and other operating systems; GCC can compile code for Android and iOS.

3.4.5 Browser

To collect information I need a browser. Information site can be opened through mobile as well as computer or laptop. Using the browser admin will add the data and change the data.

Admin panel work is done here. Admin can use any browsers like

- ❖ Google chrome
- ❖ Microsoft Edge
- ❖ Mozilla Firefox
- ❖ Brave

CHAPTER 3

Some Screenshot of running project

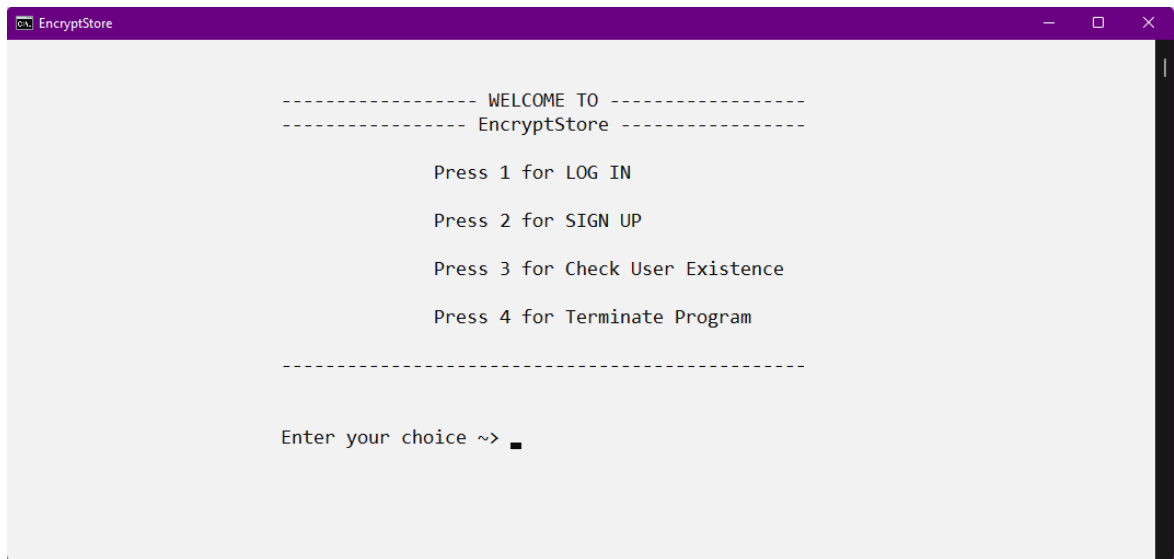


Fig.01: Homepage

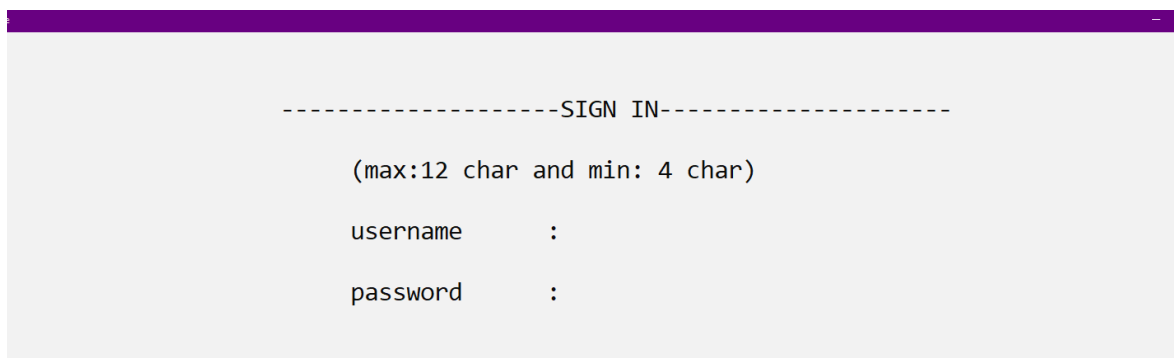


Fig.02: Sign up menu

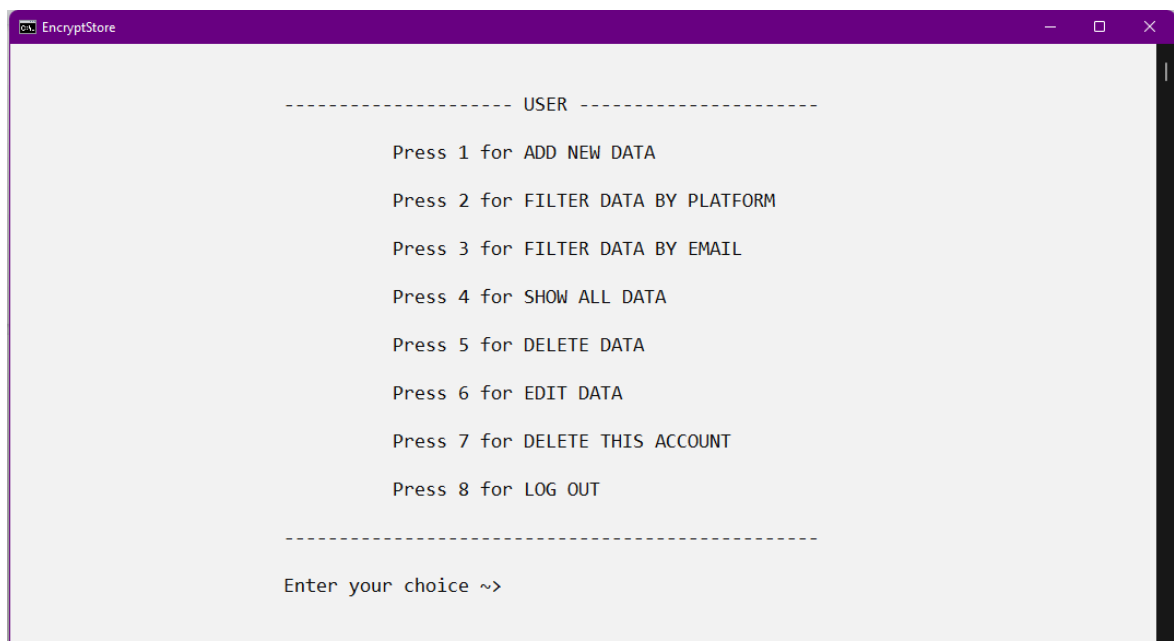
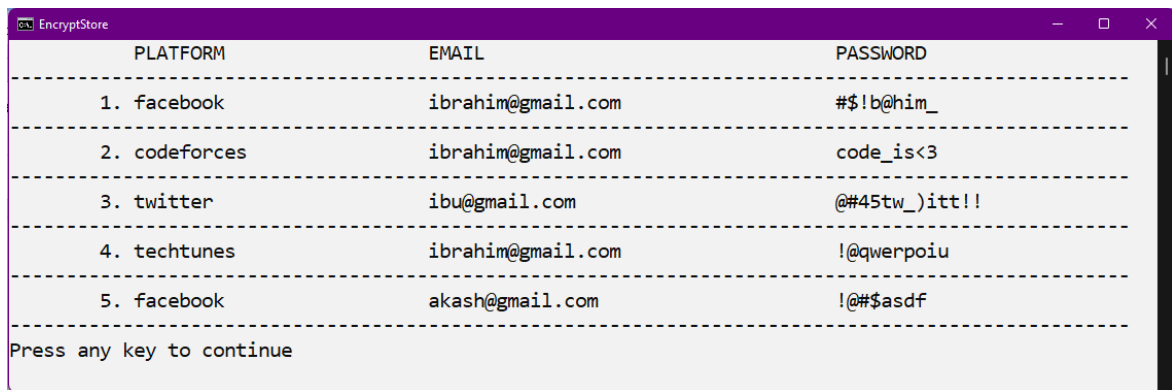


Fig.03: Logged in user menu

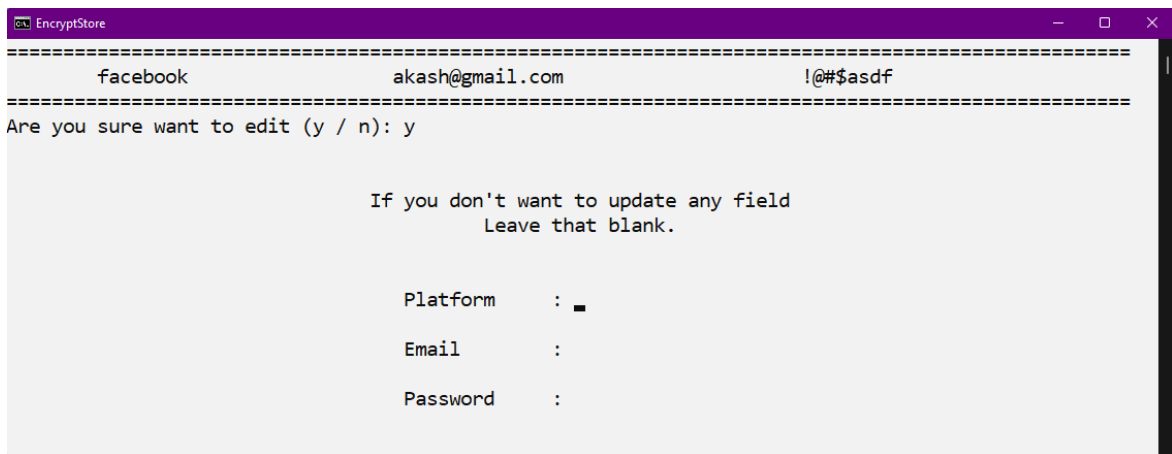


The screenshot shows a terminal window titled "EncryptStore". It displays a table with three columns: PLATFORM, EMAIL, and PASSWORD. The table contains five rows of user data. Below the table, it says "Press any key to continue".

PLATFORM	EMAIL	PASSWORD
1. facebook	ibrahim@gmail.com	#\$!b@him_
2. codeforces	ibrahim@gmail.com	code_is<3
3. twitter	ibu@gmail.com	@#45tw_)itt!!
4. techtunes	ibrahim@gmail.com	!@qwerpoi
5. facebook	akash@gmail.com	!@#\$asdf

Press any key to continue

Fig.04: List of user's data



The screenshot shows the "EncryptStore" application in edit mode. It displays the details of the user selected in the previous figure (Facebook, akash@gmail.com, !@#\$asdf). It asks "Are you sure want to edit (y / n): y". Below this, it provides instructions: "If you don't want to update any field Leave that blank." and then shows labels for Platform, Email, and Password, each followed by a colon and a cursor.

```
=====
facebook          akash@gmail.com          !@#$asdf
=====
Are you sure want to edit (y / n): y

If you don't want to update any field
Leave that blank.

Platform      : 
Email         : 
Password      :
```

Fig.05: Edit user's data

CHAPTER 5

Test Plan

In this chapter, the testing of the site and procedure of testing is depicted.

5.1 Introduction

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. The major objective of this activity is to ensure that the processing done by the application is correct and meets the objectives of the organization. Test plan aids in effective and systematic testing of the system and it aims at checking the errors of omission and commission that hinders the realization of the objectives. It takes the bottom up testing approach.

5.2 Importance of testing

Testing is used to find program errors on the system.

- ❖ It is used to find undercover errors in a program through the use of defect testing.
- ❖ Testing is also used to uncover new types of errors associated with new inventions and technology.
- ❖ Testing aims at assuring quality by enforcing consistency and reliability.
- ❖ It is used for both validation and verification to develop a product that meets user requirement.
- ❖ It is used to identify the best component combination for effective error identification.

5.3 Test Plan Strategies

The importance of the test plan is to show how the system is to be tested and also gives precise procedures to be followed during the test plan . The system will be tested with sample data to see how it would handle input and output functions as well as extreme data or conditions to determine the system behavior in overloaded situation which will directly slow the system that behaves in failure or extreme situations.

The types of testing that will be conducted upon include:

- ❖ Unit testing.
- ❖ Acceptance testing.
- ❖ System testing.
- ❖ Integration testing.
- ❖ Recovery testing.

5.3.1 Unit testing:

In this type of testing, the smallest testable parts of the system i.e. units are individually tested and independently examined for correct functionality. This type of testing involves both the positive testing and negative testing. This is important so as to make sure that the system functions properly when used both correctly and incorrectly. In this case, the forms in visual basic as well as the tables for the database will be tested individually to ensure that they are compatible. This also applies to the operating system and the software applications.

5.3.2 Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is sacrificially aimed at exposing the problems that arise from the combination of components.

5.3.3 Functional Testing:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation and user manuals.

Functional testing is centered on the following items:

- | | |
|--------------------|--|
| Valid Input | : Identified classes of valid input must be accepted. |
| Invalid Input | : Identified classes of invalid input must be rejected. |
| Functions | : Identified functions must be exercised. |
| Output | : Identified classes of application outputs must be exercised. |
| Systems/Procedures | : Interfacing systems or procedures must be exercised. |

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify business process flows, data fields, predefined processes and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current test is determined.

5.3.4 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

5.3.5 White Box Testing

White Box Testing is a testing in which the software tester has knowledge of the inner working, structure and language of the software or at least its purpose. It is used to test areas that cannot be reached from a black box level.

5.3.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structures or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source documents, such as specification or requirements document. It is a testing in which the software under test is treated as a black box. The test provides inputs and responds to outputs without considering how the software works.

5.4 Test Objectives

- ❖ To check everything correctly or not.
- ❖ If there is any error fix that.
- ❖ To minimize the chances of system failure.

5.5 Test Strategy and Approach

Field testing will be performed manually and functional tests will be written in details.

5.5.1 Features to Be Tested

- ❖ Verify that the entries are out of the correct format.
- ❖ No duplicate entries should be allowed.
- ❖ All links should take the user to the correct page.

5.5.2 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

5.6 Results:

All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 6

Limitations & Future Scope

In this chapter we summarize the limitation of this project and remarks with few directions for future works.

6.1 Limitations

- ❖ There is no online database
- ❖ There is no option for the update or insert time
- ❖ There is no backup facilities

6.2 Future work

In future I will try to add these features in my project.

- ❖ Will be implement advanced encryption system to insure user's security
- ❖ Online database system.
- ❖ Online helpline support
- ❖ Backup facilities

CHAPTER 7

Conclusion

We see that in today's world, more and more of our transactions are sent over the internet and there is nothing we can do to reverse that. And as the world is becoming a global village, it will never been more important for each and every one of us to follow safe and secure password management practices. Like I mentioned earlier Passwords are the sentry that guards our online information including bank accounts, emails, medical records and more.

I recommend that we should use password management applications to manage our accounts passwords because it's the most practical and secure way for us to follow secure password practices. Although they are not infallible and like all other software applications they are susceptible to attack, I fully do believe that it is much safer to use a password management application than not to use one. So far we have reviewed five popular account password management systems, yet, there are many others to choose from. And please take note when choosing password management software make sure it follows the principles I have covered in this study. I recommend LastPast and Keepass password manager for it has withstood the test of world top hackers. I hope by know we've understood how to better manage our account passwords and also best security practices when managing them.