Green University of Bangladesh
Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering
Semester: Fall , Year: 2023, B.Sc. in CSE (Day)
.....
Course Title: Structured Programming Lab
Course Code: CSE 104 Section: 231 (D1)
Project Report Name: Green University of Bangladesh student login Screen

1

Student Details:-
.....
Name: Promod Chandra Das
Id: 231002005
.....
Lab Date : 01/ 01/ 2024
Submission Date : 02/02/2024
Course Teacher’s Name : Humayan Kabir Rana

Contents

Chapter 1
Introduction
1.1 Background
1.2 Introduction
In the realm of software development, user authentication is a fundamental aspect of creating secure and user-friendly applications. This project centers around the design and implementation of a Login Screen using the C programming language. The objective is to provide a foundational authentication system that can be integrated into various software applications, ensuring secure access and protection of sensitive information.

As technology continues to evolve, the need for robust security measures becomes increasingly crucial. Whether in the context of standalone applications or networked systems, user

authentication forms the first line of defense against unauthorized access. This project emerges from the necessity to develop a reliable and efficient login mechanism in C, catering to diverse application domains where C remains a prevalent programming language.

Chapter 2
Project Scope
2.0.1 Objectives
The primary objectives of implementing the C-based Login Screen are as follows:

- Develop a user-friendly interface for authenticating users within C-based applications.
- Implement secure password handling and storage mechanisms to protect user credentials.
- Provide a modular and extensible codebase, allowing easy integration into various C projects.

- Ensure compatibility with different operating systems and environments.
- Offer error handling and feedback mechanisms to enhance user experience.

6

7

2.0.2 Significance of the Project

The significance of this project lies in its potential to fortify the security of C-based applications by incorporating a robust authentication system. As C remains a versatile language used in a myriad of software domains, a standardized login screen can be a valuable component for developers seeking to bolster the security of their applications. Through this project, we aim to provide not only a practical solution but also a foundation for future enhancements and integrations within the broader software development landscape. The subsequent sections of this report will delve into the methodology, system design, implementation details, testing procedures, and the anticipated impact of the C-based Login Screen in enhancing application security and user experience.

Chapter 3
System Design

3.1 Architecture

3.2 System Architecture

The architecture of the C programming Login Screen is designed to ensure a secure and efficient user authentication process. It comprises several key components that collectively contribute to the functionality and reliability of the system.

3.2.1 User Interface (UI)

The User Interface component is responsible for presenting a visually intuitive and interactive login screen to users. This includes fields

for entering the username and password, as well as buttons for submitting login credentials and handling user interactions. The UI is designed to provide clear feedback to users,
8
9
indicating successful or unsuccessful login attempts.

3.2.2 Authentication Module

The Authentication Module is the core component responsible for verifying the user's identity based on the provided credentials. This module typically includes functions for validating usernames and passwords, checking against a stored database of user credentials, and enforcing security measures such as account lockouts after multiple failed login attempts.

3.2.3 Password Handling

To enhance security, the Password Handling component is dedicated to securely storing and managing user passwords. Passwords are often hashed using cryptographic algorithms to protect sensitive information from unauthorized access. Additionally, the system may include mechanisms for password recovery or reset in case users forget their credentials.

```
\documentclass[12pt, a4paper]{report}

% Packages
\usepackage{geometry}
\usepackage{graphicx}
\usepackage{titlesec}
\usepackage{fancyhdr}
\usepackage{lipsum} % For generating dummy text, you can remove it later

% Page setup
\geometry{a4paper, margin=2.5cm}
\pagestyle{fancy}
\fancyhf{}
\rhead{\thepage}
\renewcommand{\headrulewidth}{0pt}
```

```
% Title setup

\titleformat{\chapter}[display]

  {\normalfont\huge\bfseries}{\chaptertitlename\ \thechapter}{20pt}{\Huge}


% Document begins

\begin{document}


% Title page

\begin{titlepage}




  \centering

  \vspace*{\fill}

  \Huge \textbf{Green University of Bangladesh    \\Department of Computer Science and Engineering (CSE) \\ Faculty of Sciences and Engineering \\Semester: Fall , Year: 2023, B.Sc. in CSE (Day)

  \\.....\\Course Title: Structured Programming Lab\\Course Code: CSE 104 Section: 231 (D1) \\Project Report Name: Green University of Bangladesh student login Screen  }


  \vspace*{\fill}

\end{titlepage}


\Huge \textbf { Student Details:- \\.....\\Name: Promod Chandra Das \\ Id: 231002005\\..... }


\Huge \textbf {Lab Date : 01/ 01/ 2024 \\ Submission Date : 02/02/2024 \\ Course Teacher’s Name : Humayan Kabir Rana }
```

% Table of contents

\tableofcontents

% List of figures and tables (if needed)

% \listoffigures

% \listoftables

% Chapters

\chapter{Introduction}

\section{Background}

% Your content here

\section{Introduction}

In the realm of software development, user authentication is a fundamental aspect of creating secure and user-friendly applications. This project centers around the design and implementation of a Login Screen using the C programming language. The objective is to provide a foundational authentication system that can be integrated into various software applications, ensuring secure access and protection of sensitive information.

As technology continues to evolve, the need for robust security measures becomes increasingly crucial. Whether in the context of standalone applications or networked systems, user authentication forms the first line of defense against unauthorized access. This project emerges from the necessity to develop a reliable and efficient login mechanism in C, catering to diverse application domains where C remains a prevalent programming language.

\chapter{Project Scope}

\subsection{Objectives}

% Your content here

The primary objectives of implementing the C-based Login Screen are as follows:

\begin{itemize}

\item Develop a user-friendly interface for authenticating users within C-based applications.

\item Implement secure password handling and storage mechanisms to protect user credentials.

\item Provide a modular and extensible codebase, allowing easy integration into various C projects.

\item Ensure compatibility with different operating systems and environments.

\item Offer error handling and feedback mechanisms to enhance user experience.

\end{itemize}

\subsection{\textbf{Significance of the Project}}

The significance of this project lies in its potential to fortify the security of C-based applications by incorporating a robust authentication system. As C remains a versatile language used in a myriad of software domains, a standardized login screen can be a valuable component for developers seeking to bolster the security of their applications. Through this project, we aim to provide not only a practical solution but also a foundation for future enhancements and integrations within the broader software development landscape.

The subsequent sections of this report will delve into the methodology, system design, implementation details, testing procedures, and the anticipated impact of the C-based Login Screen in enhancing application security and user experience.

\Huge \textbf { }

\chapter{System Design}

\section{Architecture}

\section{System Architecture}

The architecture of the C programming Login Screen is designed to ensure a secure and efficient user authentication process. It comprises several key components that collectively contribute to the functionality and reliability of the system.

\subsection{User Interface (UI)}

The User Interface component is responsible for presenting a visually intuitive and interactive login screen to users. This includes fields for entering the username and password, as well as buttons for submitting login credentials and handling user interactions. The UI is designed to provide clear feedback to users, indicating successful or unsuccessful login attempts.

\subsection{Authentication Module}

The Authentication Module is the core component responsible for verifying the user's identity based on the provided credentials. This module typically includes functions for validating usernames and passwords, checking against a stored database of user credentials, and enforcing security measures such as account lockouts after multiple failed login attempts.

\subsection{Password Handling}

To enhance security, the Password Handling component is dedicated to securely storing and managing user passwords. Passwords are often hashed using cryptographic algorithms to protect sensitive information from unauthorized access. Additionally, the system may include mechanisms for password recovery or reset in case users forget their credentials.

\subsection{Error Handling and Logging}

Effective error handling is crucial for providing users with meaningful feedback and diagnosing issues within the system. The Error Handling and Logging component is responsible for capturing and logging error messages, which can aid in debugging and improving the system's overall reliability.

\subsection{User Management}

The User Management component oversees tasks related to user account creation, modification, and deletion. It may include functionality for user account registration, profile updates, and the enforcement of account policies (e.g., password complexity requirements).

\subsection{Security Measures}

The Security Measures component includes additional layers of security to protect against common threats, such as brute-force attacks and session hijacking. This may involve implementing account lockout mechanisms, session timeouts, and secure communication protocols.

\subsection{Integration with External Systems}

In some cases, the C programming Login Screen may need to integrate with external systems, such as databases or directory services, to retrieve user information and ensure consistency across the application environment.

\subsection{Flow of Execution}

The typical flow of execution involves user interaction with the UI, submission of login credentials, validation through the Authentication Module, and subsequent actions based on the success or failure of the authentication process.

The effectiveness of the architecture lies in its ability to seamlessly integrate these components, providing a secure and user-friendly login experience for applications developed in the C programming language.

The subsequent sections will delve into the detailed design and implementation considerations for each of these architectural components.

% Your content here

\Huge \textbf { }

\chapter{Implementation}

\section{Technologies Used}

% Your content here

\Huge \textbf {C Programming Language:

The entire implementation is written in the C programming language, utilizing its syntax, data types, control structures, and functions.

\section {Standard Input/Output (stdio.h):}

The printf and scanf functions from the standard input/output library (stdio.h) are used for displaying messages to the user and obtaining input from the user.

\section {Standard Library (stdlib.h):}

The stdlib.h library is used for functions like fgets to read a line from the standard input and atoi to convert strings to integers.

\section {String Handling (string.h):}

The string.h library is used for string manipulation functions like strcmp to compare strings and strcspn to remove trailing newline characters.

\section {Data Structures:}

The implementation uses a simple data structure, the struct, to define the User structure for storing username and password information.

\section {Loops and Control Structures:}

Standard loops (while loop) and control structures (if statement) are used for managing the login attempts and authentication process.}

\chapter{Testing and Validation}

\section{Test Cases}

% Your content here

\Huge \textbf {Valid Login Credentials:

\section {Input: Correct username and password.

Expected Result: Successful login.

Invalid Username:

Input: Incorrect username and correct password.

Expected Result: Unsuccessful login with an appropriate error message.

Invalid Password:

Input: Correct username and incorrect password.

Expected Result: Unsuccessful login with an appropriate error message.

Empty Username:

Input: Empty username and correct password.

Expected Result: Unsuccessful login with an appropriate error message.

Empty Password:

Input: Correct username and empty password.

Expected Result: Unsuccessful login with an appropriate error message.

Exceed Maximum Login Attempts:

Input: Incorrect username or password for more than the allowed attempts.

Expected Result: Account locked message after reaching the maximum allowed attempts.

Case Sensitivity:

Input: Mixed case username and correct password.

Expected Result: System handles case sensitivity appropriately.

Whitespace Handling:

Input: Username or password with leading or trailing whitespaces.

Expected Result: System trims whitespaces and performs the login.

Special Characters in Username/Password:

Input: Username or password with special characters.

Expected Result: System handles special characters appropriately.

Security Checks:

Input: Attempt to bypass login by directly modifying memory or manipulating variables.

Expected Result: System remains secure, and any such attempts are detected and prevented.

Concurrency Issues:

Input: Simultaneous login attempts from multiple users.

Expected Result: System handles concurrency gracefully and maintains data integrity.

User Management:

Input: Attempt to log in with a username that does not exist.

Expected Result: Unsuccessful login with an appropriate error message. }}

\chapter{Conclusion}

\section{Summary}

% Your content here

\Huge \textbf { While serving as
a foundational example of user au-
thentication, the implementation may
benefit from future enhancements,
such as incorporating encryption for
password handling and exploring in-
tegration with external databases.
}

% Appendices (if needed)

```
% \appendix
% \chapter{Additional Information}

% Bibliography (if needed)
% \bibliographystyle{plain}
% \bibliography{references}

\end{document}
```