BGS POLYTECHNIC

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

A CAPSTONE PROJECT PRESENTATION ON

"PASSWORD STRENGTH CHECKER WITH ENCRYPTION"

CHORT OWNER :-

Mrs. ARCHANA V Be

TEAM MEMBERS:-

P SAI ESWAR REDDY (498CS22073)

SHUBHASHREE R (498CS22077)

SAHANA D L (498CS22072)

SHREYAS K M (498CS22075)

Password Strength Checker with Encryption

Cyber Security Firm

TABLE OF CONTENTS

INTRODUCTION

The project focuses on developing a **password strength checker** that evaluates password security based on complexity, and length.

The project integrates **intelligent password analysis** with advanced encryption techniques to enhance security.

The project enhances **password security** by ensuring strong, encrypted authentication mechanisms.

The project's significance goes beyond theoretical research to address **real-world cybersecurity challenges** in authentication.

OBJECTIVES

- **Check password strength:** Develop a tool that checks the strength of a password based on a set of predefined rules, such as length, complexity, and character set.
- **Encrypt passwords:** Encrypt the passwords using a secure encryption algorithm, such as AES or RSA, to protect them from unauthorized access.
- **Provide feedback:** Provide feedback to the user on the strength of their password, including suggestions for improvement.
- **Store encrypted passwords:** Store the encrypted passwords securely, using a secure storage mechanism such as a hashed password database.
- **Develop a user interface:** Develop a user-friendly interface that allows users to input their password and receive feedback on its strength.

LITERATURE SUREVY

[1]. Kumar, Singh, and Patel (2020) conducted a study that evaluated traditional password strength checkers and introduced the integration of encryption techniques to enhance security. Their work highlighted the limitations of rule-based checkers, which focused on length and complexity but often failed to account for common patterns. The authors proposed using a hybrid encryption method combining Advanced Encryption Standard (AES) and RSA, ensuring that passwords are encrypted as they are evaluated.

[2]. Li, Wang, and Chen (2021) introduced an AI-based approach to password strength evaluation combined with advanced encryption methods. Their study utilized deep learning models trained on vast datasets of leaked password patterns to predict potential vulnerabilities in newly created passwords.

[3]. Brown, Lee, and Gonzalez (2022) explored end-to-end encryption for password strength checkers in highly sensitive environments, such as banking and healthcare. Their research integrated password strength evaluations with Elliptic Curve Cryptography (ECC), which provided stronger encryption with smaller keys compared to RSA.

[4]. Martinez, Rodriguez, and Zhang (2023) proposed the use of quantum-resistant cryptography in password strength checkers to counteract future threats posed by quantum computing.

[5]. Nguyen, Kim, and Sharma (2024) investigated password strength checkers integrated with homomorphic encryption to provide privacy-preserving password strength assessments.

METHODOLOGY

- **Requirement Analysis Define project goals:** assess password strength, provide feedback, and ensure encryption-based security.
- **Design and Planning:** Create UI/UX wireframes and plan system architecture integrating frontend validation and AES encryption.
- **Development Frontend:** Build the interface using HTML, CSS, and JavaScript.
- **Password Strength Checker:** Implement logic to evaluate length, character types, and entropy.
- **Encryption:** Use JavaScript-based AES encryption to secure passwords before storing or transmitting.
- **Testing:** Conduct unit testing for strength checker logic and AES functions. Perform usability and compatibility testing across browsers.

PROBLEM IDENTIFICATION

- Users create weak passwords due to lack of awareness.
- Passwords are often stored without encryption, risking data breaches.
- Existing strength checkers lack integrated encryption and comprehensive feedback.
- Sensitive data is vulnerable during storage and transmission.

DELIVERABLES

- **Password strength checker tool:** A tool that checks the strength of a password based on a set of predefined rules.
- **Encryption module:** A module that encrypts the passwords using a secure encryption algorithm.
- **Feedback mechanism:** A mechanism that provides feedback to the user on the strength of their password.
- **Secure storage mechanism:** A mechanism that stores the encrypted passwords securely.
- **User interface:** A user-friendly interface that allows users to input their password and receive feedback on its strength.
- **Documentation:** Documentation that explains how to use the tool, including instructions for users and administrators.
- **Test report:** A test report that summarizes the results of testing the tool, including any issues or bugs that were identified.

DESCRIPTION

- A web tool to check password strength in real-time.
- Built with HTML, CSS, and Java Script. Evaluates passwords based on length, character variety, and complexity.
- Uses AES encryption to secure passwords after checking.
- Helps users create strong, secure passwords.

KEY MILESTONE

- Project planning and tech stack selection
- UI design for password input and feedback
- Implement password strength checker logic
- Add AES encryption functionality
- Test and debug the application
- Write documentation and reports
- Prepare final presentation and demo
- Submit completed project

CBS COST BREAKDOWN STRUCTURE

SL NO	DESCRIPTION	COST
1	**Hardware Costs**	₹ 4,000.00
2	**Software Costs**	₹ 8,000.00
3	**Licensing and Subscription Fees**	₹ 1,000.00
4	**Infrastructure Costs**	₹ 2,000.00
5	**Testing and Evaluation Costs**	₹ 6,000.00
6	**Documentation and Reporting Costs**	₹ 7,000.00
7	**Training & Miscellaneous Costs**	₹ 2,000.00
TOTAL COST		**₹ 30,000.00**

TIMELINE SCHEDULE

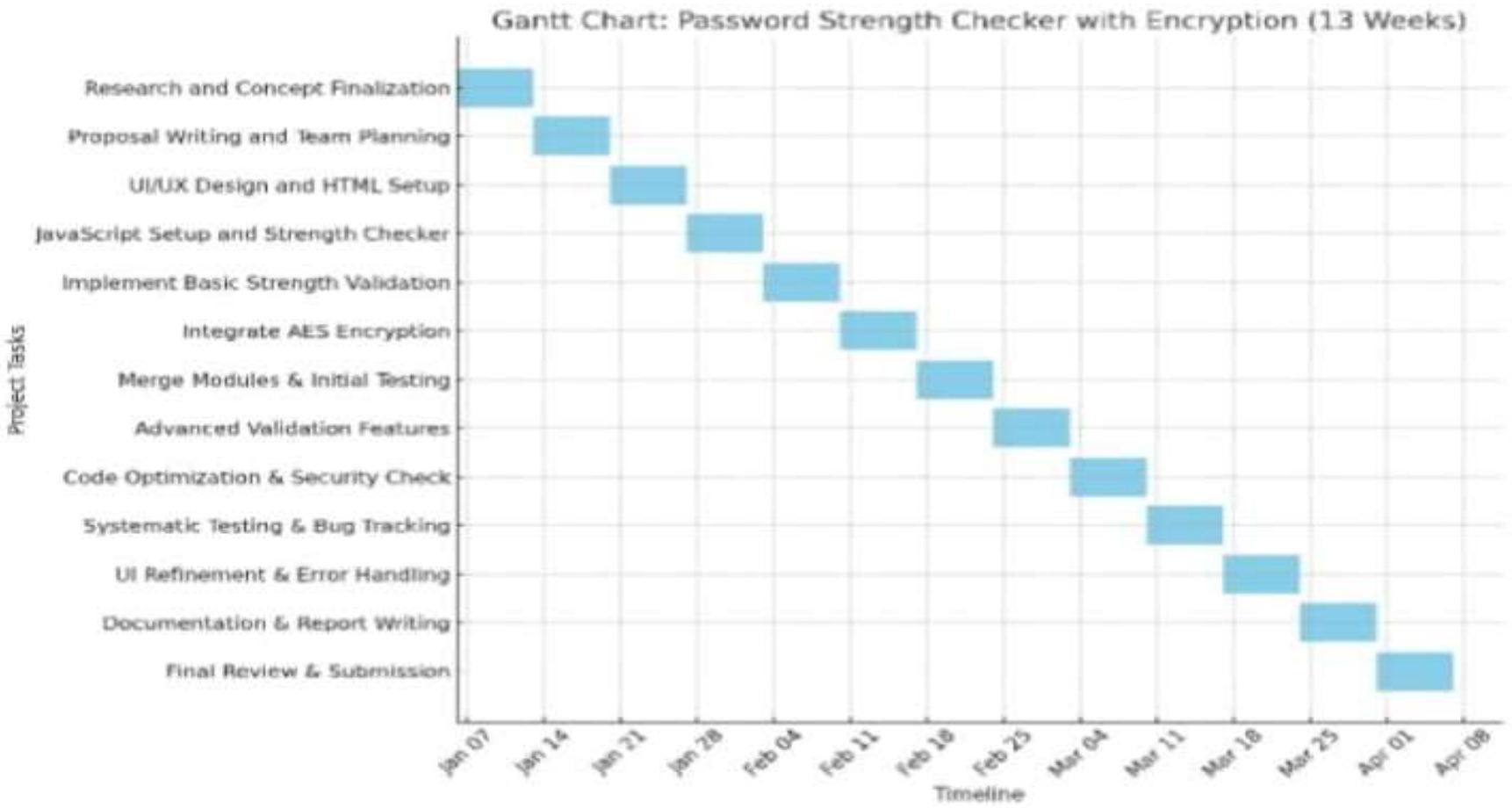


Fig:- Timeline Schedule

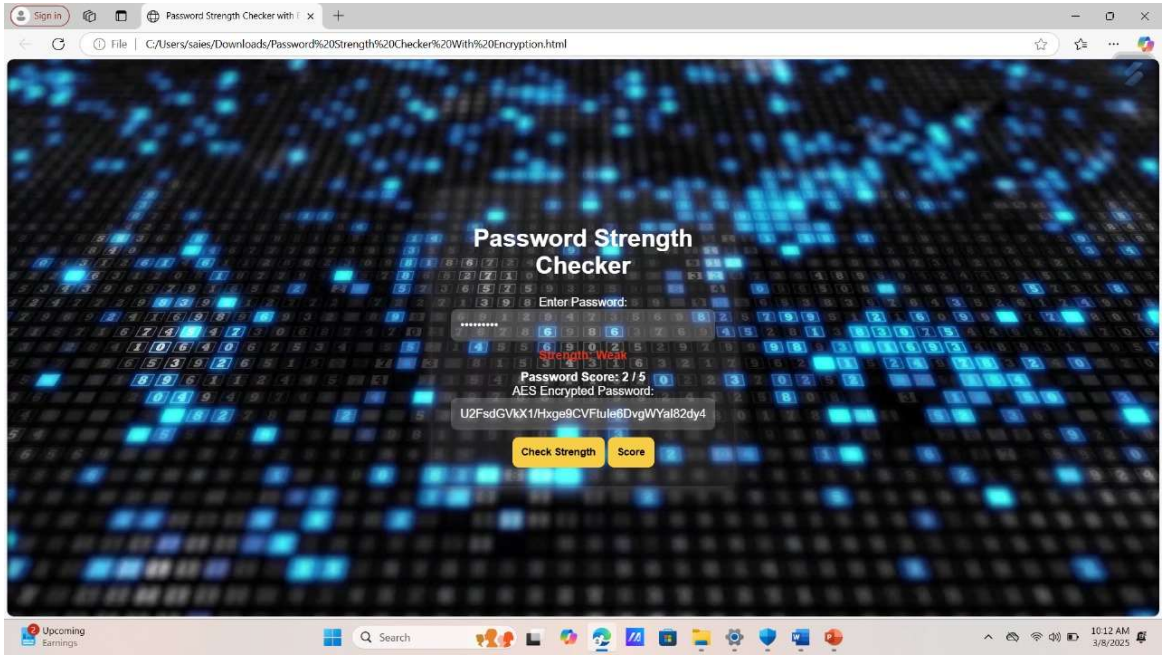
RISK ANALYSIS

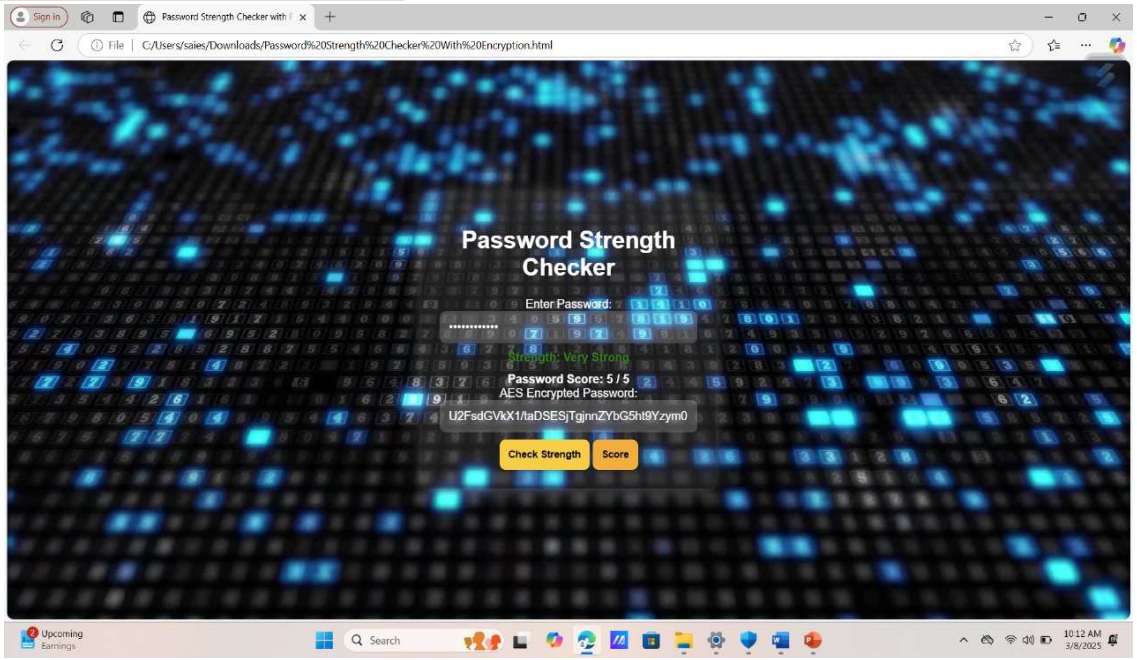
- AES encryption errors can compromise data.
- Browser issues may affect tool performance.
- Weak evaluation logic can mislead users.
- Key exposure and XSS attacks are major security risks.
- Delays in testing/documentation can impact delivery.
- User confusion and trust issues may reduce effectiveness.

TESTING PHASES

- **Unit Testing –** Test individual functions.
- **Integration Testing –** Ensure components work together.
- **Functional Testing –** Validate password checks and output.
- **Security Testing** – Check AES encryption and input safety.
- **Browser Testing –** Verify across multiple browsers.
- **User Testing –** Get feedback on usability.
- **Performance Testing –** Test speed and efficiency.

RESULTS


Password Strength Checker with
C:/Users/saies/Downloads/Password%20Strength%20Checker%20With%20Encryption.html
Password Strength Checker
Enter Password:
Strength: Weak
Password Score: 2 / 5
AES Encrypted Password:
U2FsdGVkX1/Hxge9CVFtule6DvgWYal82dy4
Check Strength
Score


Password Strength Checker with
C:/Users/saies/Downloads/Password%20Strength%20Checker%20With%20Encryption.html
Password Strength Checker
Enter Password:
Strength: Very Strong
Password Score: 5 / 5
AES Encrypted Password:
U2FsdGVkX1/taDSESjTgjnnZYbG5ht9Yzym0
Check Strength
Score

SYSTEM ARCHITECTURE

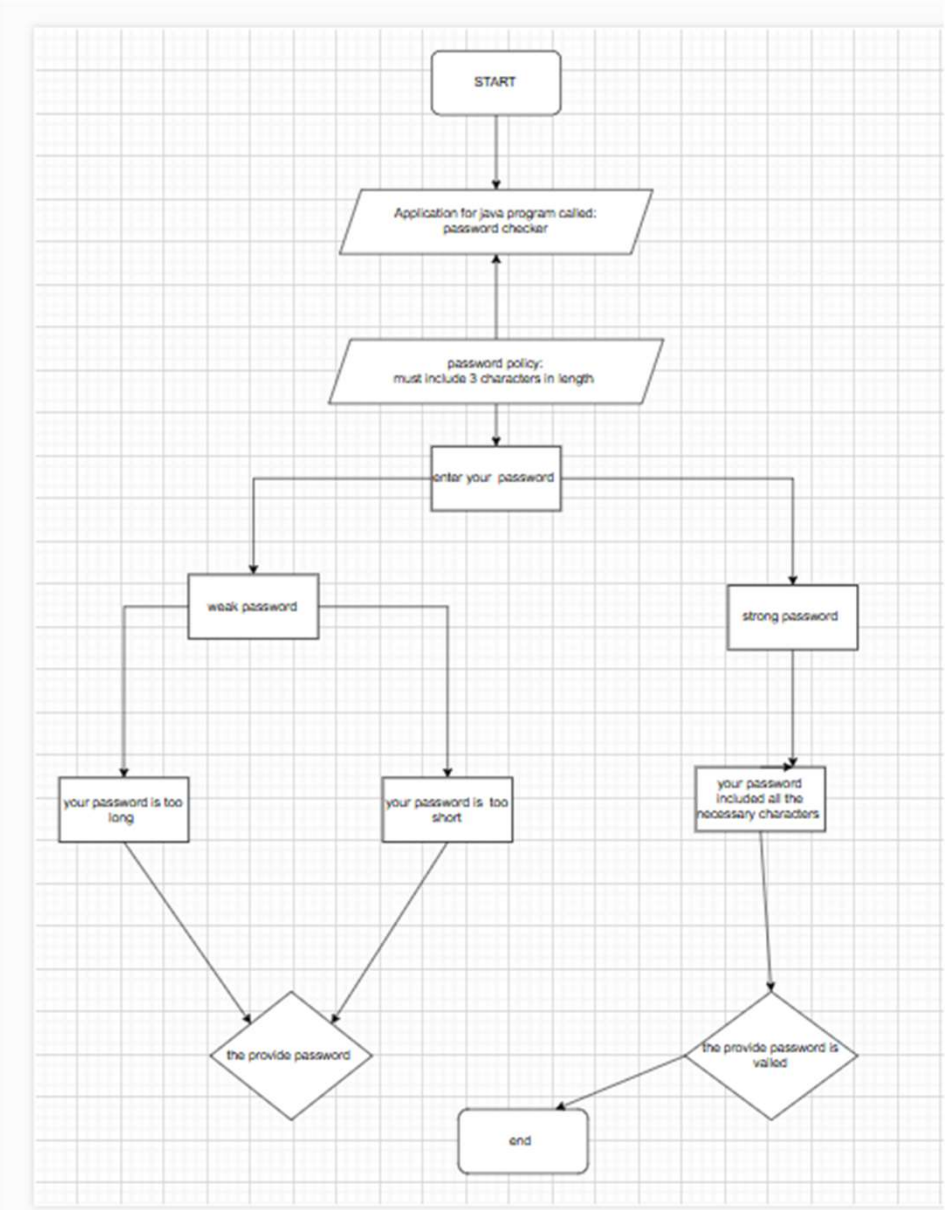


Fig:- System Architecture

DATAFLOW DIAGRAM

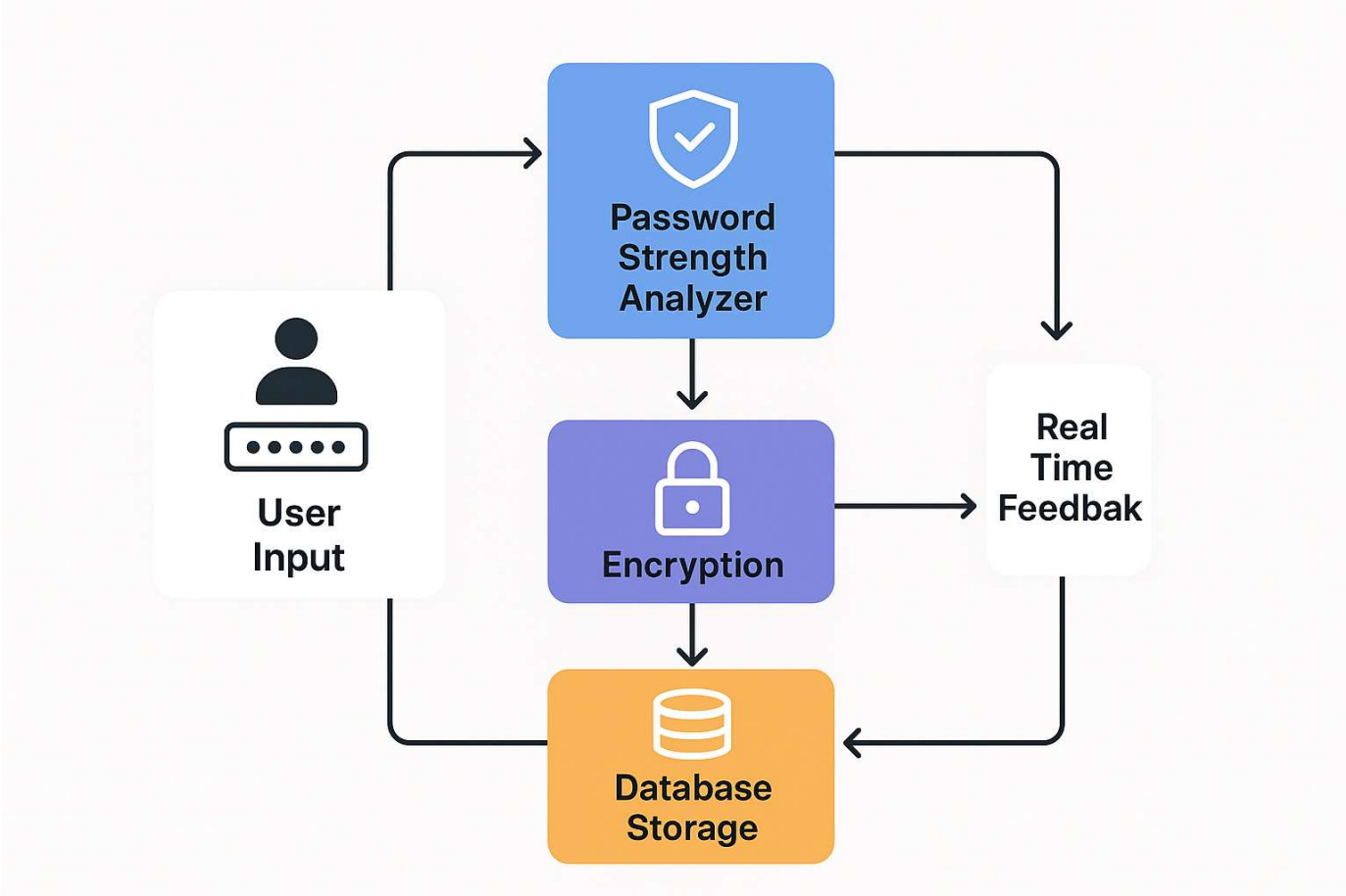


Fig:- Dataflow Diagram

CONCLUSION

Applies cutting-edge techniques to enhance encryption mechanisms and data protection.

Focuses on how encrypted communication functions in diverse IOT environments.

Uses intelligent models to strengthen encryption based on evolving threats.

Highlights importance of secure encryption protocols to defend against data breaches and cyberattacks.

Lays the groundwork for developing more advanced and secure encryption technologies.

FUTURE SCOPE

- **AI-Based Password Analysis**
- **Multi-Factor Authentication (MFA)**
- **User Interaction and Feedback**
- **Enhanced Encryption Mechanisms**


THANK YOU