Secure ATM System

Purpose and Features Overview

Purpose of the Software

- Highlight the primary goal: Provide a secure, user-friendly ATM system.
- Mention the target audience: Bank customers and administrators.
- State the problem it solves: Prevent unauthorized access and secure transactions.

Key Features

- Secure User Registration (password hashing).
- Multi-Factor Authentication (MFA).
- User Roles: Administrator and Customer.
- Real-time balance updates and transaction management.

User Functions

- Sign Up and Login.
- Deposit and Withdraw Money.
- Check Account Balance.
- Admin-specific functions: View user activity or perform audits.

Security Measures in Features

- Passwords hashed with SHA-256.
- Input sanitization to prevent SQL injection.
- MFA for enhanced login security.
- Database encryption at rest (optional for further improvement).

SDLC Phases and Security Measures

Software Development Life Cycle (SDLC) and Security Measures

SDLC Phases and Security Measures

- Software Development Life Cycle (SDLC) and Security Measures
- Mention the SDLC phases:
  - Requirements Gathering
  - Design
  - Implementation
  - Testing
  - Deployment & Maintenance.
- Briefly explain the importance of addressing security at each phase.

Security in Requirements Gathering

- Understanding potential risks:
  - User authentication.
  - Sensitive data protection (balance, passwords).
- Identifying compliance requirements (e.g., GDPR).

Security in Design Phase

- Incorporating OWASP guidelines.
- Designing secure database schemas: Avoiding SQL injection risks.
- Data flow analysis for identifying attack surfaces.

Security in Testing and Deployment

- Testing:
  - Penetration tests to find vulnerabilities.
  - Secure code reviews.
- Deployment:
  - Secure connection to the database using SSL.
  - Server hardening and monitoring.

Implemented Security Features and Benefits

Security Features in Action

Multi-Factor Authentication (MFA)

- Explain MFA workflow:
    - Random code generation.
    - User verification using the code.
- Benefit: Reduces risks of unauthorized access.

Password Hashing with SHA-256

- Why hashing is important: Protects passwords in case of database leaks.
- How SHA-256 ensures strong one-way encryption.
- Benefit: Safeguards user credentials effectively.

SQL Injection Prevention

- Use of prepared statements for all database queries.
- Benefit: Eliminates common SQL injection vulnerabilities.

End-to-End Security Benefits

- Improved user trust due to robust security measures.
- Protection of financial data and accounts.
- Compliance with modern security standards.