Experiment 23:

Security Measures for a Mobile App

Authentication & Authorization

- **Implement OAuth 2.0 or JWT (JSON Web Tokens) for secure authentication.**
- **Use multi-factor authentication (MFA) for enhanced security.**
- **Enforce strong password policies with encryption.**

Data Protection & Encryption

- **Encrypt sensitive data using AES-256 for storage and TLS 1.2/1.3 for transmission.**

PAGE 1 →

Secure API Communication

- **Use HTTPS with SSL/TLS for secure API calls.**
- **Implement certificate pinning to prevent MITM (Man-in-the-Middle) attacks.**
- **Validate API requests with rate limiting and authentication.**

Code Security & Protection

- **Obfuscate and minify code to prevent reverse engineering.**
- **Use secure coding practices to prevent vulnerabilities like SQL injection and XSS.**

← PAGE 2 →

Secure Storage & Permissions

- **Store sensitive data in encrypted databases instead of local storage.**
- **Restrict unnecessary app permissions to reduce exposure.**
- **Prevent screenshots of sensitive screens using FLAG_SECURE (Android).**

Session Management

- **Implement automatic session timeouts and token expiration.**
- **Use secure cookies with HTTPOnly and Secure flags.**

← PAGE 3 →

Fraud & Threat Detection

- **Monitor user behavior for anomalies and fraud detection.**
- **Integrate real-time threat intelligence and monitoring tools.**
- **Use CAPTCHA to prevent automated bot attacks.**

Regular Security Updates

- **Keep third-party libraries updated to patch vulnerabilities.**
- **Perform regular penetration testing and code audits.**

← PAGE 4