PHASE 6

SECURITY RECOMENDATIONS

**Submitted to:**

Nick Hamnett, Instructor (CPRG 303-F)

School of Advanced Digital Technology, SAIT

**Submitted by:**

Paramveer Singh

SAIT ID 000891084

Mobile Application Development, CPRG 303-F

School of Advanced Digital Technology, SAIT

**A**. **Security Recommendation**: Implement robust input validation and sanitation. This is chosen because it's a fundamental security measure that applies to virtually all mobile applications. Proper input validation prevents a wide range of common vulnerabilities like SQL injection, cross-site scripting, and buffer overflows, which can compromise both user data and app integrity. This recommendation is particularly crucial for apps that interact with external servers or APIs, process user input, or handle sensitive data. It's a foundational security practice that ensures only properly formatted data is processed by your application.

**B**. **Beneficiaries**: Robust input validation and sanitation primarily benefit the end-users of the app by protecting them from potential security breaches and data leaks. It also benefits developers and the organization behind the app by preventing vulnerabilities that could lead to exploitation, maintaining the integrity and trustworthiness of the app, and potentially avoiding legal or reputational repercussions associated with security incidents.

**C**. **Source**: This recommendation is based on the [Android App Security Checklist](https://github.com/muellerberndt/android_app_security_checklist), specifically from the sections dealing with data storage and cryptography.

**D**. **Implementation Timeline**:   
Implementing secure client-server communication using HTTPS with SSL/TLS should be a priority from the very beginning of the app's development process. Given the serious security risks associated with unsecured data transmission, such as data breaches or man-in-the-middle attacks, it's crucial to integrate this security measure early. This will help ensure that all data transmitted between the app and the server is encrypted and secure right from the initial stages, safeguarding user data and the integrity of the app throughout its lifecycle.

**E**. **Necessity**: Tic Tac Toe game app needs secure client-server communication due to several factors: First, if we storing player data and game states on a server, this data needs to be transmitted securely to prevent unauthorized access or manipulation. Second, considering your consideration of hardware components like fingerprint scanners, any biometric data or authentication details must be transmitted securely to maintain user privacy and trust. Essentially, secure communication is vital to protect user data and ensure a reliable, trustworthy gaming experience.

**F**. **Application and Feasibility**:

* **Application**: Integrating SSL/TLS in the app's networking code. Ensure all requests to the server are made over HTTPS. This includes data transmissions for game states, player profiles, and any interaction with backend services.
* **Feasibility**: The implementation is highly feasible. Most modern networking libraries and frameworks support HTTPS out of the box, making it relatively straightforward to set up. The primary requirement is obtaining a valid SSL certificate for your server, which is a standard practice and widely supported. This approach ensures that all data transmitted between your app and server is encrypted and secure, aligning with best practices for app security.