BAB V
PENUTUP

5.1 Simpulan

Berdasarkan hasil uji coba *penetration testing* yang dilakukan pada website yang menjadi tugas akhir saya, berikut adalah beberapa kesimpulan yang dapat diambil:

1. Peningkatan risiko akses tidak sah pada halaman login: Celah *SQL Injection* memungkinkan penyerang mencuri kredensial dan memanipulasi database, yang dapat mengakibatkan pengambilalihan akun dengan hak istimewa atau modifikasi data sensitif. Solusinya adalah implementasikan query berparameter atau prepared statement untuk semua interaksi database. Hindari penggunaan query yang langsung menyisipkan input pengguna tanpa validasi.

2. Kurangnya proteksi pada fitur *chat* terhadap *XSS*: Fitur chat tidak memiliki mekanisme *escape* input yang memadai, sehingga memungkinkan skrip berbahaya disisipkan dan dieksekusi oleh browser pengguna lain, membahayakan data pengguna dan memfasilitasi penyusupan *malware*. Solusinya adalah Gunakan mekanisme encoding input pengguna dengan fungsi seperti htmlspecialchars() untuk memastikan input aman sebelum dirender di browser.

3. Eksploitasi referensi objek langsung pada fitur transaksi: Fitur transaksi rentan terhadap manipulasi parameter melalui *IDOR*, memungkinkan

penyerang mengakses atau mengubah data transaksi tanpa otorisasi, termasuk saldo pengguna lain. Solusinya adalah Gunakan pengenal unik global *(GUID)* untuk referensi objek alih-alih parameter langsung seperti username atau ID. Pastikan adanya validasi otorisasi di server sebelum memproses permintaan terkait objek sensitif.

4. Kebutuhan mendesak untuk validasi dan *sanitasi input*: Setiap fitur aplikasi menunjukkan kelemahan dalam validasi *input*, baik itu untuk mencegah *SQL Injection*, *HTML*, maupun skrip *XSS*, yang menunjukkan bahwa mekanisme sanitasi input belum diimplementasikan secara konsisten. Solusinya adalah Terapkan strategi sanitasi input secara menyeluruh dengan daftar putih *(whitelist)* karakter yang diizinkan. Gunakan framework keamanan seperti Laravel Validation untuk membangun lapisan validasi yang konsisten di seluruh aplikasi.

Dengan demikian, hasil dari uji coba penetration testing ini menunjukkan adanya beberapa celah keamanan yang dapat dimanfaatkan oleh pihak tidak bertanggung jawab. Oleh karena itu, langkah-langkah perbaikan yang telah direkomendasikan perlu segera diterapkan untuk meningkatkan keamanan sistem, melindungi data pengguna, dan memastikan integritas serta ketersediaan aplikasi.

5.2 Saran

Laporan hasil *penetration testing* ini dapat ditingkatkan dan dimodifikasi agar lebih bermanfaat bagi penguji lain yang ingin memperluas cakupan pengujian dengan pendekatan yang lebih mendalam.

Selain menggunakan metode yang sudah dilakukan pada proyek tugas akhir ini, disarankan untuk mencoba pendekatan *grey-box* testing dengan memberikan akses terbatas ke struktur database atau sebagian kode aplikasi. Misalnya, penguji dapat diberikan kredensial akun pengguna biasa untuk mengidentifikasi potensi eskalasi hak akses. Selain itu, metode *fuzz testing* dapat diterapkan dengan menggunakan alat seperti *AFL (American Fuzzy Lop*) untuk mengidentifikasi input yang menyebabkan perilaku aplikasi yang tidak terduga. Analisis statis kode juga bisa dilakukan dengan alat seperti *SonarQube* untuk mendeteksi celah keamanan pada tahap pengembangan.

Dengan memperluas dan menyesuaikan metode pengujian ini, laporan ini dapat menjadi pedoman yang lebih kaya dan praktis bagi penguji lainnya untuk terus meningkatkan keamanan aplikasi.