Beurteilungsraster Projekt Modul 183

Kriterium	Doku-Eintrag zwingend	Indikatoren	Belegende Feststellungen	-	0	+
Allgemein	**doppelt gewichtet (0.4)**					
Die gemäss Projektantrag definierten Anforderungen sind funktionsfähig implementiert	-	Applikation stürzt nicht ab, Funktionen funktionieren wie gewünscht gem. Projektantrag (ohne Sicherheitsbetrachtung)				
sicheres Sessionhandling						
Funktionen, welche nur bestimmte Benutzer / Benutzergruppen ausführen dürfen, sind sowohl im Frontend wie auch im Backend berücksichtigt und funktionieren	-	Wenn der Benutzer das Recht nicht hat: * Funktion nicht sichtbar * Funktion nicht manuell mit Tools auf dem Server aufrufbar * allfällige Formulare werden gar nicht ausgeliefert				
Der Login-Mechanismus ist sicher implementiert, Passwörter sind gehashed und gesalzen gespeichert	ja Wie wurde das realisiert?	Login gemäss erarbeitetem Ablauf, Passwörter mit Salt und Hash in der DB, aktuell als offiziell sicher erachteter Hashalgorithmus wird verwendet.				
Das Problem der Session-Fixation wurde in der Lösung berücksichtigt und abgesichert	ja Wie wurde das realisiert?	Invalidierung der Session zum geeigneten Zeitpunkt. Allfällige Übernahme von Daten aus der alten Session.				
Absicherung Standardangriffe						
Injections werden durch entsprechende Gegenmassnahmen verhindert	ja welche Massnahmen wurden an welchen Stellen getroffen	Prepared Statements, Named Queries mit Parametern, Inputvalidierung (Client+Server), Rechtebeschränkung DB-User				
XSS-Angriffe werden durch entsprechende Gegenmassnahmen verhindert	ja welche Massnahmen wurden an welchen Stellen getroffen	Inputvalidierung (Client+Server) Output Escaping sichere Konfiguration der Session/Sessioncookie (**C**ontent **S**ecurity **P**olicy -> evt. als Schwerpunkt wählbar)				
Errorhandling, Logging						
Der Benutzer erhält keine Fehlermeldungen, welche auf Applikationsinterna schliessen lassen. Dies erfordert ein adäquates Errorhandling und Logging	-	Keine Exceptions gelangen bis zum Client Eigene Fehlerseiten für HTTP Stati 403, 404 und 500 -> auch über Unterschied dev/prod Umgebung möglich Fehler/Ereignisse werden intern adäquat geloggt				
Eigenes Kriterium mit eigenem Fokusthema	**doppelt gewichtet (0.4)**					
Beispiel: CSRF Angriffe werden durch entsprechende Gegenmassnahmen verhindert	ja wie wurde es implementiert, weiterführende Gedanken	Berücksichtigung eines Tokens Begrenzung der Sessiondauer Formulare per POST übertragen Bei Stateless: Double Submit Pattern				
				0	0	0

Startnote:	4
Bewertungsschritt:	0.2
Note:	**4**

Bedeutung Bewertungsstufe

- \- Kriterium nicht ersichtlich / nicht erfüllt, Doku fehlt oder ist sehr lückenhaft und knapp
- 0 Kriterium teilweise erfüllt / Doku vorhanden und Kernaussagen korrekt dokumentiert
- \+ Kriterium erfüllt / Doku ausführlich und stimmig