Module 5- Computer Systems (2022-23)

Project

UNIVERSITY OF TWENTE.

Testing-Security by Design Checklist

Team ID: 4	**Team Members:** Thijs Frauenfelder, Katy Radzkova, Ayolt ten Have, Victor Zugravu, Frank Bosman, Mikus Vancans
Project Name: PiSec	**Mentor(s):** Priya Naguine & Radu Basarabá

Instructions:

1. Refer to the below table. All the mentioned points are mandatory to perform for your application except point no. 4.
2. You should consider atleast 2 vulnerabilities for each criteria given in Column 'B', except point no. 4, 6, and 7.
3. The mitigation plan/solution should be considered for every identified vulnerability.
4. Make sure to review the document with your team members and mentor(s) before final submission.
5. This checklist should be in lined and submitted along with the Software Testing document.

Points	Source Code Review, Static and Dynamic Application Testing	Identified Vulnerabilities for testing (Name them)	Put tick ✔(if you have completed all the points as mentioned in Column 1.	Remarks, if any
1	Application security vulnerabilities (e.g. Access Control, Injection, Authentication, Cross Site scripting, etc.)	sql injection to access unauthorized data and cross site scripting to run malicious code on the web application.	✔	-
2	Weak security in functions (e.g. old encryption techniques, Hashing, Privileges assigned, Function error, etc.)	old encryption or hash functions could already be broken and not be secure anymore.	✔	-
3	Duplicate/unnecessary functions	Duplicate functions could be called while they have been changed already and unnecessary functions makes maintaining the application harder.	✔	-
4	Analyzing Program (e.g. computation time, power consumption, etc.) **(Optional)**	-		-
5	Address the remaining vulnerabilities of your application (manual)	destruction of the physical application. Put something in front of the camera.	✔	-
6	Make a mitigation plan/solution by listing down the vulnerabilities	**mitigation plan** **sql injection:** use of input sanitisation and prepared statements **cross site scripting:** filter user input. **old encryption functions:** don't use old encryption functions or self created ones but use validated encryption functions. **hash functions:** Make use of sha-512 for hashing. **Duplicate functions:** Remove duplicate functions if they exist. **Unnecessary functions:** Remove functions which are not used or called.	✔	-

		Destruction of physical application: Built a case around the security system for protection. **Putting something in front of the camera:** The user should remove it when noticed.		
7	Review with your team members and approve by your mentor(s).	-	✔	-

Team members reviewed: (Frank Bosman, Yes), (Victor Zugravu, Yes), (Mikus Vancans, Yes), (Thijs Frauenfelder, Yes), (Katy Radzkova, Yes), (Ayolt ten Have, Yes)

Mentor(s) reviewed and verified: (Priya Naguine, yes), (Radu Basarabá , …)

Prepared by:

Dipti K. Sarmah (Project Coordinator)