Module 5- Computer Systems (2022-23)

Project

UNIVERSITY OF TWENTE.

Software Testing Document (STD) Template

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

Instructions:

1. Refer to the below table and complete all the sections with clarity.

2. Select those test strategies that are applicable to test your application.

3. Make sure to refer to the "Development-Security by Design Checklist" to see the possible vulnerabilities in your application.

4. Feel free to add features and test cases in the table that are essential to test your application.

5. You can use Selenium, SonarQube, and/or GitLab CI/CD to perform source code review, static and dynamic application testing, etc.

Test Strategy	Date (When did you perform the testing?)	Process/Function (Features to be tested)	Test Case	Step	Description	Status (Passed/Failed /Open)	Expected Results	Actual Result	Mitigation plan/Solutions	Review on the Mitigation plan (Passed/Failed)	Remarks on the Failed mitigation plan
Unit testing	1-11-2022	Database	correct storage database	1	unit test if the database stores account data correctly	**Passed**	account data is stored correctly according to the unit test	account data is stored correctly according to the unit test			
Manual testing	21-10-2022	Detection	Motion detection	1	When there is motion the motion detector gives a signal.	**Passed**	When there is motion detected a signal is received	When there is motion detected a signal is received			
				2	the alert information is stored in database after detection	**Passed**	Alert information is stored in the database after motion detection	Alert information is stored in the database after motion detection			
				3	notification is sent to final user after detection	**Passed**	an email and whatsapp notification is sent after motion detection	notifications are sent after motion is detected.			
Application testing	31-10-2022	Authentication	Logging in	1	Correct login details are filled in	**Passed**	The user is logged in and redirected to his personal home page	The user is logged in and redirected to his personal home page			
Application testing	31-10-2022	Security	SQL Injections	2	Preventing malicious SQL injections	**Failed**	SQL injections are prevented	SQL injections are not prevented	i) Check if code of input sanitisation is implemented correctly ii) Check if prepared statements for	**Passed**	

							queries are used.		

Note: Refer to the following documentation on GitLab and SonarQube for clarity-

1. Source Code review with SonarQube: **https://docs.sonarqube.org/latest/**

2. GitLab integration with SonarQube: **https://docs.sonarqube.org/latest/analysis/gitlab-integration/**

3. SonarQube (Static Application Testing):
   https://www.sonarqube.org/features/security/

4. Gitlab (Static Application Testing):
   https://docs.gitlab.com/ee/user/application_security/sast/

5. GitLab (Dynamic Application Testing):
   https://docs.gitlab.com/ee/user/application_security/dast/

Prepared by:

Dipti K. Sarmah (Project Coordinator)