CYBER SECURITY IN AUTOMOTIVE TRAINING PROGRAM

1. **Objective:** In order to train students with theoretical and practical knowledge about automotive cyber security in LGE VS Company. After completing the program, a student will:

- Understand the lifecycle of software development in automotive domain
- Have knowledge about some cyber security features and work scopes in automotive domain
- Have experience in hardware setup, feature implementation and cyber security quality assurance

2. **Trainees**: Final year students

3. **Duration**: 3 months with Full time students

4. **Method**:

- **Overview Theory**: Given lectures by Professors
- **Projects**: proceeded by Student Teams(2 members)

5. **Content**:

No.	Scope	Session	Expected Outcome	Duration (Days)	Materials	Training Method	Mentor
1	Automotive & Technical Overview	1. Automotive Domain Overview		1	**File**: AUTOMOTIVE Overview.pdf	Giving **Lectures** by DUT Professor	N/A
		2. Cyber Security in Automotive	Understanding cyber security hacking cases, cyber security regulation and technology for	1	**Video**: VS Automotive Cyber security Awareness Improvement Education		

			product development				
		3. Software Engineering	Understand basic SW engineering topics before joining actual task.	2	**LinkedIn**: Git Essential Training: The Basics **Videos**: - Understanding of Static Analysis and MISRA Rules - Overview of IPC Linux Programming - Threads synchronization - Automotive SPICE in Practice		
2	Way of work	4. Scrum based Way of Work	Apply Scrum Framework to do the projects	1	**Website**: Home | Scrum.org **File**: Scrum Way Of Work.pdf		
3	Environment Setup	5. Environment setup	Complete a system with: - Target system: Raspberry PI 4 - Host system: window x86/Linux x86 - SW platform and framework: WebOS Open Source Edition.	5 Sprints (10 weeks)	**File:** Environment Setup.pdf	Doing **Project** by Student Team Apply Scrum (File: Scrum Way Of Work.pdf)	DUT Professor
4	Development	6. Firewall	Define the whitelist for all software components. Apply the rule for incoming / outgoing traffic.		**Folder**: Firewall		

		7. SMACK	Define the Access Control Rule for all software components. Apply the rule for read / write access permission for each component.		**Folder**: SMACK
5	Security Features Verification	8. Security Features Test	Design/Execute Security Test cases for implemented features (Smack, Firewall)		**Files**: • 12.2 Smack Test Specification_v1 • 12.1 Task and Test case Example_v1 **Tool:** Scapy
		9. Open Source Software Vulnerability Scan (OSSVS)	Can analyze open-source software to find related CVEs, analyze CVEs and apply patches.		**Overview:** Securing the Software Supply Chain **Tool:** • OWASP Dependency Check • https://pypi.org/project/fosslight-yocto/
		10. Exploit Memory(EM)/ 11.Operational Security Hardening (OSH)	EM: Understand memory protection measures and apply them to an executable. OSH: Understand system weaknesses and mitigate them using Bash scripts.		**Overview:** https://opensource.com/article/21/6/linux-checksec **Tool:** checksec.sh

6	Final Report	Presentation & Evaluation		5		Giving **Presentation** by Student Team	DUT Professor LGE DV Specialist
TOTAL				**12 weeks**			